



HTML *javascript*

CSS PHP DOM

ZÁKLADY TVORBY WWW STRÁNEK
A JEDNODUCHÝCH WWW APLIKACÍ

Studijní text

Kristýna Kubrická
Jan Kubrický

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

OBSAH

OBSAH	2
ÚVOD	4
1 ÚVOD DO STUDIA	6
INTERNET	6
PRINCIP ČINNOSTI SLUŽBY WWW	7
HTML.....	7
INTERNETOVÝ PROHLÍŽEČ.....	8
HTTP.....	8
URL.....	9
W3C A NORMY	10
JAK TO TEDY CELÉ PROBÍHÁ	10
SHRNUTÍ KAPITOLY	12
SAMOSTATNĚ.....	13
2 ZÁKLADY JAZYKA HTML	14
XHTML.....	14
ZÁKLADY JAZYKA XHTML.....	15
TAGY	16
LOGICKÉ ROZDĚLENÍ	16
ZÁSADY A KONVENCE JAZYKA XHTML.....	17
NEJPOUŽÍVAJŠÍ TAGY	20
ENTITY.....	21
OBRÁZKY A HYPERTEXTOVÉ ODKAZY.....	22
MULTIMEDIÁLNÍ OBJEKTY	25
NÁPOVĚDA PS PADU A ONLINE ZDROJE	26
CO PŘINÁŠÍ HTML5.....	27
PRO ZÁJEMCE.....	30
SHRNUTÍ	30
3 KASKÁDOVÉ STYLY	32
CSS	32
TŘI ZPŮSOBY ZÁPISU CSS	34
CSS – SYNTAXE	36
LAYOUT WWW STRÁNEK.....	38
CSS3	41
SHRNUTÍ KAPITOLY	42
SAMOSTATNĚ.....	43
4 HTML A CSS	44
ŠABLONY.....	44
TVORBA VLASTNÍ ŠABLONY.....	46
KÓDOVÁNÍ	46
DĚDIČNOST STYLŮ	50
SHRNUTÍ	52
SAMOSTATNĚ.....	53
5 WEBOVÉ FORMULÁŘE	54



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

ZÁKLADY FORMULÁŘŮ	54
FORMULÁŘE V PRAXI	55
PRVKY FORMULÁŘE	57
SLUŽBY TŘETÍCH STRAN	57
SHRNUTÍ	58
SAMOSTATNĚ.....	58
6 JAVASCRIPT	59
ÚVOD.....	59
POUŽITÍ	60
VLOŽENÍ JAVASCRIPTU DO STRÁNKY	61
MOŽNOSTI ZÁPISU JAVASCRIPTU V HTML DOKUMENTU	61
PRVKY A SYNTAXE JAZYKA	62
OBJEKTOVÉ VLASTNOSTI JAVASCRIPTU	68
DOCUMENT OBJECT MODEL (DOM)	70
KNIHOVNY JAVASCRIPTU	72
SHRNUTÍ	73
7 ÚVOD DO JAZYKA PHP	75
JAZYK PHP	75
PRAKTICKÉ PŘÍKLADY A UKÁZKY	76
JEDNODUCHÉ PŘÍKLADY.....	76
KONSTRUKCE	80
SHRNUTÍ	84
SAMOSTATNĚ.....	84
PŘÍKLADY	85
8 OPTIMALIZACE (SEO)	86
SEO.....	86
DEFINICE.....	87
METODY SEO.....	87
NETICKÉ METODY	89
SHRNUTÍ	90
SAMOSTATNĚ.....	90
9 PUBLIKOVÁNÍ WWW	92
WEBHOSTING	92
FTP	94
POUŽITÁ LITERATURA	96
PŘÍLOHA 1	97
APACHE, PHP, MYSQL.....	97
XAMPP	97
KOŘENOVÝ ADRESÁŘ PRO DOKUMENTY	100
PROBLÉMY	101



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Úvod

Služba WWW prošla za poslední desetiletí bouřlivým vývojem. Od statických prezentací až po multimediální projekty v podobě sociálních sítí a informačních portálů. Na těchto projektech se většinou podílí velké skupiny odborníků z různých oblastí webových technologií, ale rovněž marketingu, komunikace, obchodování atd.

Tvorba jednoduchých www stránek je ale běžným uživatelům poměrně blízko. Na trhu je k dispozici množství webových editorů, tzv. WISIWYG (co vidíš, to dostaneš), které umožňují tvořit webové stránky v podstatě každému. Připomeňme například MS Publisher, Expression Web 2, Web Page Maker nebo již pokročilejší Adobe DreamViewer. V podstatě bychom mohli říci, že jste-li schopni vytvořit textový dokument v textovém procesoru, jste schopni vytvořit webovou stránku ve webovém editoru.

Tento studijním text je ale více zaměřen přímo na představení a popis **základních technologií** www stránek a sleduje jejich tvorbu *zápisem zdrojového kódu*. Předváděny jsou tak elementární návrhy statických šablon www stránek prostřednictvím jazyků (X)HTML a CSS. Tj. podíváme se to, co provádí na pozadí výše zmíněné webové WISIWYG editory automaticky.

Je k takovému studiu vzhledem k existenci webových editorů vůbec důvod? Dobrá otázka. Chcete-li se v tvorbě www stránek posunovat neustále dále, tak ano, je. Navíc i při sebelepším webovém editoru se tvůrce webových stránek zásahu do zdrojového kódu nevyhne. A zde přichází na řadu znalosti spojené s tvorbou a editací zdrojového kódu. Navíc čistota (kvalita) zdrojového kódu generovaná webovými editory je stále na hony vzdálená ručnímu psaní zdrojového kódu uživatelem, čili vámi.

Pokročilejší, interaktivní webové stránky neboli webové aplikace se vyznačují různými funkčními částmi. Ovšem při hlubším pohledu nám neunikne, že obsahují některé společné prvky, jako jsou emailové formuláře, výpis aktuálních informací nebo možnosti editace záznamů uživatelem. Z množství programovacích technologií jsou probírány skriptovací jazyky JavaScript a PHP. **Cílem je naučit se vytvářet webové aplikace za pomoci právě těchto technologií.** Studijní opora je zaměřena na prezentaci jak hotových ukázkových řešení, tak typických situací, které vývojáři webových aplikací řeší dnes a denně.

Podmínky studia

Úspěšné studium je podmíněno základními znalostmi práce s:

- operačním systémem Windows, Linux nebo Mac,
- pokročilým textovým editorem,
- libovolným internetovým prohlížečem.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Studijní opora obsahuje také přiloženou složku **ZIP** (cvicne-soubory.zip), ve které naleznete zdrojové kódy probíraných příkladů. Dále ukázkové soubory příkladů využití PHP a JavaScriptu. **Soubory obsahují důležité komentáře!** Můžete si je přímo stáhnout z odkazu v poznámce pod čarou.¹

Studijní text rovněž obsahuje externí odkazy na on-line výukové zdroje. Pevně věřím, že po prostudování budete sami schopni vytvářet webové stránky, které budou sloužit dobře nejen vám, ale i vašim návštěvníkům.

Důležité!

Tento studijní text neobsahuje některé důležité pasáže – zejména věnující se objektově orientovanému programování v jazyce PHP a v roce 2016 bude inovována. Proto si tuto problematiku musíte částečně nastudovat z jiných zdrojů. Závěrečná práce (zdrojové kódy) obsahuje základy použití, které byste měli ovládat.

Za všechny gramatické chyby, překlepy, případně i faktické chyby se omlouváme. Budeme Vám velmi vděční, pokud nás na ně upozorníte na email jan.kubricky@upol.cz.

Autoři

¹ <http://www.kteiv.upol.cz/frvs/ict-kubricky/?page=TWWW/cvicne-priklady.zip>

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

1 Úvod do studia

Cíle

Po prostudování této kapitoly budete:

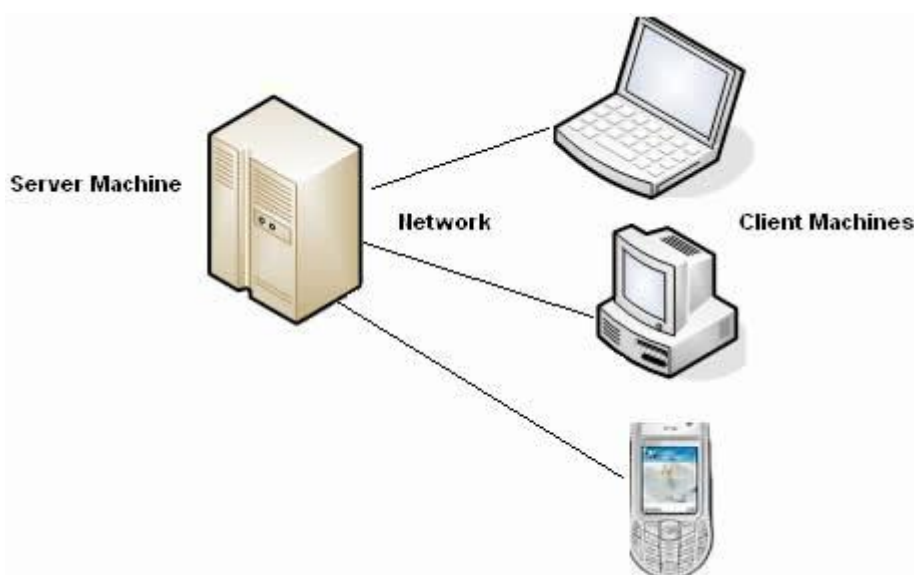
- umět definovat pojem Internet,
- znát základní služby internetu,
- umět definovat a popsat vztah mezi pojmy WWW, HTML, HTTP a URL,
- seznámeni s principem přenosu informací prostřednictvím služby WWW.

Doba potřebná ke studiu

Na prostudování kapitoly Vám bude stačit **60 minut**.

Internet

Internet je systém, který propojuje počítačové sítě po celém světě, v nichž jednotlivé **připojené počítače komunikují prostřednictvím sady protokolů TCP/IP**. Počítače v síti Internet pracují buď jako **servery**, nebo jako klientské stroje (**klienti**). Jejich význam je zřejmý: servery své služby poskytují, klientské počítače tyto služby využívají.²



² Jak funguje INTERNET: názorné video, doporučuji shlédnout:
<http://www.stream.cz/uservideo/128119-jak-funguje-internet>

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Myšlenka celosvětové sítě se zrodila ve vědeckých kruzích a na půdě univerzit poté, co se s úspěchem zřizovalo propojení sítí v rámci instituce NASA a dalších vybraných středisek. Prvním významným krokem k rozšíření Internetu mezi nejširší veřejnost bylo zavedení služby **World Wide Web**, kterou dnes známe pod označením **WWW**. Tato přelomová služba vznikla roku 1989 ve Švýcarsku, v evropském centru pro jaderný výzkum CERN. Její vynálezce Tim Bernes Lee ji původně zamýšlel jako vnitropodnikový systém. O tom do jaké míry se rozšířila, svědčí i slovo World v jejím názvu. Je ovšem chybou se domnívat, že Internet je sítí pouze jedné služby. Po krátké úvaze vás jistě napadne minimálně jedna další.

Mezi nejznámější a nejvíce využívané patří zejména:

- **E-mail** – elektronická pošta zajišťující přenos zpráv.
- **FTP** – přenos souborů mezi FTP serverem a FTP klientem.
- **On-line komunikace** – textová, hlasová i obrazová komunikace kdekoli na světě (ICQ, Skype, internetové televize a rádia).

Princip činnosti služby WWW

WWW je služba internetu, celosvětová pavučina, která propojuje systém hypertextových dokumentů napsaných podle určitých pravidel a umožňuje jejich přenos. K tvorbě dokumentů pro WWW (www stránek) je nutná existence vývojového nástroje, který umožňuje formátovat text, vložení hypertextových odkazů, obrázků, zvukových stop atp. To vše má na starosti od svého vzniku v roce 1991 značkovací jazyk www stránek, **jazyk HTML**.

Pozn.: Hypertextové dokumenty – dokumenty obsahující hypertextové odkazy umožňující vzájemné propojení těchto dokumentů do funkčního celku.

HTML

HTML (Hypertext Markup Language), slouží k označení a popisu všeho, co má být na www stránce umístěno. WWW stránky jsou pak jednoduše vzato souborem jednoho nebo více hypertextových dokumentů ve formátu HTML, které jsou uspořádány a propojeny tak, že tvoří funkční celek.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický



INTERNETOVÝ PROHLÍŽEČ

Aby bylo možné beze zbytku využít potenciálu služby WWW a prohlížet hypertextové dokumenty, vznikly speciální programy, tzv. internetové prohlížeče (browsersy), které **umožňují komunikaci s http (webovými) servery a jsou schopny jazyk HTML zpracovat a přeložit**, tj. uživateli zobrazit webovou stránku. Mezi nejznámější internetové prohlížeče patří:

- Internet Explorer od společnosti Microsoft
- Mozilla Firefox
- Opera
- Google Chrome od společnosti Google
- Safari a další.



HTTP

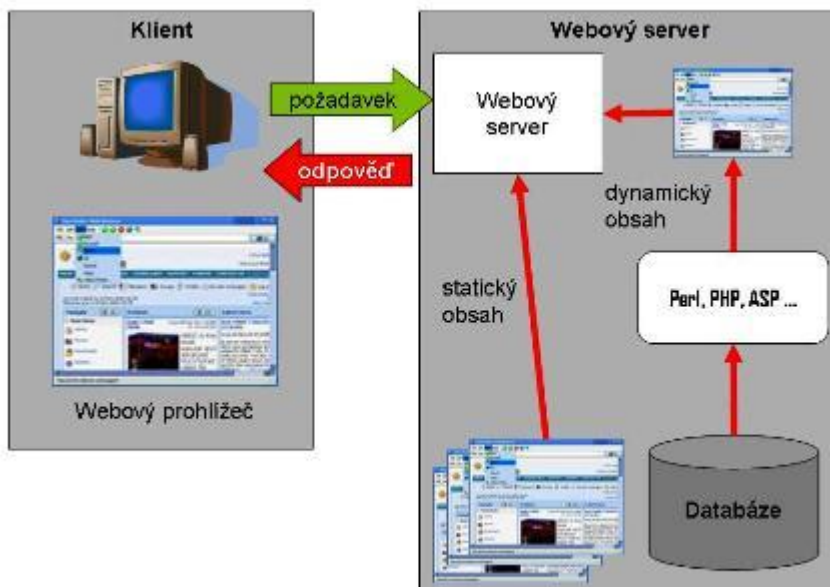
V prostředí sítě internet pracují počítače buď jako servery nebo jako klienti. Server poskytuje služby klientům, klienti služeb serveru využívají. V rámci služby WWW hovoříme konkrétně o webovém (HTTP) serveru. Tyto servery slouží jako uložení www stránek a zpracovávají požadavky, které k nim vyšlou uživatelé prostřednictvím internetových prohlížečů. **Klient musí serveru zaslat požadavek prostřednictvím určitých pravidel.** Jestliže server požadavku rozumí, je schopen ho zpracovat a dokáže podle akce uživatele odpovědět, zašle klientovi opět dle zmíněných pravidel příslušnou odpověď (např. www stránku, dokument ke stažení atd.) (viz následující obrázek). Pravidla této komunikace určuje internetový protokol HTTP (Hypertext Transfer Protocol).

HTTPS - Bezpečnější varianta protokolu, která umožňuje přenášená data šifrovat a tím chránit před odposlechem či jiným narušením.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický



Obr 1: Komunikace klient/web server

(zdroj: Wikipedia. Webový server. http://cs.wikipedia.org/wiki/Webový_server#mediaviewer/Soubor:Webserver.jpg)

URL

URL, celým názvem Uniform Resource Locator („jednotný lokátor zdrojů“) je řetězec znaků s definovanou strukturou, který slouží k přesné specifikaci umístění zdrojů informací (ve smyslu dokument nebo služba) na Internetu. **URL definuje doménovou adresu serveru, umístění zdroje na serveru a protokol, kterým je možné zdroj zpřístupnit.**

Příklad URL:

<http://www.kteiv.upol.cz/frvs/ict-kubricky/>

Pozn.: URL obsahuje ještě další nepovinné části, které jsou buď předdefinovány nebo nemají význam. Více o nich na závěr této kapitoly.

Doménová adresa serveru je vlastně jeho přeložená IP adresa, které je pro lepší zapamatování přiděleno DNS serverem doménové jméno. Určitě se shodneme, že se lépe pamatuje URL ve tvaru

<http://www.movz.upol.cz/>

než

<http://158.194.63.8/>



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Co je to **IP adresa** a jaká je její funkce a význam v prostředí Internetu v této části nebudeme vysvětlovat, ale k tomuto tématu si jistě dopředu najdete spoustu informací sami.

W3C a Normy

World Wide Web Consortium (W3C) je mezinárodní konsorcium, jehož členové společně s veřejností vyvíjejí webové standardy pro World Wide Web. Cílem konsorcia je „Rozvíjet World Wide Web do jeho plného potenciálu vývojem protokolů a směrnic, které zajistí dlouhodobý růst Webu“. W3C se také zabývá vzděláním a přístupností, vyvíjí software a nabízí otevřenou diskuzi o Webu prostřednictvím fóra. Konsorciu předsedá jeho zakladatel Tim Berners-Lee, tvůrce služby Webu a primární autor specifikací URL (Uniform Resource Locator), HTTP (HyperText Transfer Protocol) a HTML (HyperText Markup Language) - základních pilířů Webu.

Mnoho formálních norem, další technické specifikace a software definuje operace různých aspektů World Wide Webu, Internetu a serverů pro výměnu informací. Mnoho z těchto dokumentů je prací World Wide Web Consortium (W3C).

- Doporučení pro značkovací jazyk, hlavně **HTML** a **XHTML** od W3C. Definují strukturu a interpretaci hypertextových dokumentů.
- Doporučení pro styly, především **CSS** od W3C.
- Normy pro ECMAScript (obvykle forma **JavaScriptu**).
- Doporučení pro **Document Object Model (DOM)** od W3C.

Jak to tedy celé probíhá

Většina webových aktivit začíná na straně klienta, a to okamžikem, kdy uživatel spustí svůj webový prohlížeč. Tento prohlížeč ze všeho nejdříve načte dokument takzvané domovské stránky, která se může nacházet buďto na lokálním disku (u vás na počítači), nebo častěji na serveru některé ze sítí - (třeba Internetu www.seznam.cz).

Pokud je domovská stránka umístěna mimo vlastní počítač, musí se klientský prohlížeč nejprve „poradit“ **se serverem systému pojmenování domén** (se serverem DNS, Domain Name System) a převést jméno serveru s dokumentem domovské stránky, např. www.seznam.cz, na jeho IP adresu; teprve poté může odeslat tomuto serveru přes Internet příslušný požadavek.

Webový server stráví většinu svého času nasloucháním sítě a čekáním na požadavek, ve kterém je uvedena jeho vlastní jedinečná adresa. Jakmile se objeví takovýto požadavek, server odešle požadovaný dokument do prohlížeče žadatele. Zároveň server obvykle zaznamená jak požadavek, tak i jméno klientského počítače, požadovaný dokument a časový okamžik. Vrátime se zpět do

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

prohlížeče, kam mezitím dorazil vyžádaný dokument. Pokud je to textový soubor v prostém formátu ASCII, zobrazí jej většina prohlížečů také v prostém, textovém formátu.

Prohlížeče načítají ze serverů také **binární soubory**. Stažený binární soubor však prohlížeč obvykle uloží přímo na lokální disk, kde s ním uživatel může dále pracovat; pokud je ale na počítači nainstalován odpovídající pomocný program, případně speciální zásuvný (plug-in) software či applet, může se zvuk či videosoubor (filmový klip) zobrazit přímo, respektive se začne přímo přehrávat.

Ve většině případů prohlížeč dostane speciální dokument, který je zdánlivě prostým textovým souborem, ale ve skutečnosti obsahuje kromě textu také speciální označovací kódy, jimž se říká tagy (tags). **To je dokument HTML, XHTML nebo XML**; prohlížeč jej odpovídajícím způsobem zpracuje, text naformátuje podle informací definovaných tagy, a stáhne případné speciální, doplňující soubory, jako jsou například obrázky.

Celý proces si ještě na závěr této části jednoduše zrekapitulujeme. Vývojář vytvoří www stránky, které umístí na webový server. Pro přenos souborů na server se používá protokolu (služby) **FTP** (File Transfer Protocol), konkrétně například aplikace **FileZilla** nebo **Total Commander**, které práci s FTP podporují.

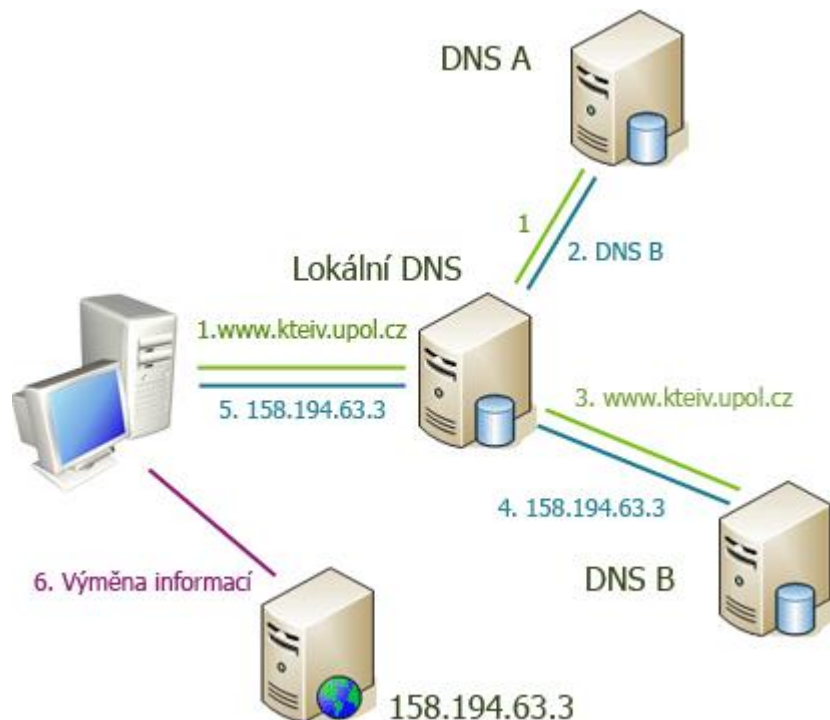
Webový server je v síti internet jednoznačně identifikován adresou IP, které může DNS server přidělit několik doménových jmen. Na jednom serveru tak můžou hostovat www stránky Katedry technické a informační výchovy (www.kteiv.upol.cz) i stránky projektu IVOŠ (ivos.upol.cz). Organizaci zobrazení jednotlivých stránek při požadavku uživatele si pak dle svého nastavení řídí sám webový server.

Zadáme ve svém internetovém prohlížeči požadavek ve formě URL <http://www.kteiv.upol.cz> (viz. krok 1 na obr. 2). Nejprve se ověří existence webového serveru. To zajišťují DNS servery (spravují databáze elektronických adres). Lokální DNS server vyšle požadavek na server DNS A. V případě že DNS server A doménové jméno ve své databázi neobsahuje, vrátí odkaz na další DNS server, který by odpověď znát mohl (krok 2 a 3). Když je existence hledaného webového serveru ověřena (krok 4 a 5), pošle náš počítač tomuto serveru požadavek na stránku. Server ověří, zda takový dokument existuje, a jestliže ano, okamžitě ho pošle na náš počítač a stránka se nám v internetovém prohlížeči zobrazí (krok 6).

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický



Obr. 2 Zpracování požadavku na www stránku v síti Internet

Pozn.: Řešení dotazu a princip činnosti DNS serverů je v praxi trochu složitější než v uvedeném popisu, nicméně základní postup je našemu podání velmi blízký.

Shrnutí kapitoly

- Internet je celosvětová „sítí sítí“, jejímž prostřednictvím lze využívat službu WWW.
- Počítače v síti Internet pracují jako servery anebo jako klientské stroje. Jejich význam je zřejmý: servery své služby poskytují, klientské počítače tyto služby využívají.
- Služba WWW je postavena na aplikačním protokolu HTTP a komunikaci server - klient.
- Díky internetovým prohlížečům můžeme prohlížet obsah www stránek, jejichž základ je tvořen značkovacím jazykem HTML nebo jeho variantami (XHTML).
- Zvládnutí HTML je základní podmínkou pro postup do dalších úrovní programování na webu a je nutné pro používání složitějších skriptovacích jazyků, jako je například PHP, ASP či Java.
- Dotazy, které prostřednictvím internetových prohlížečů do sítě vysíláme, jsou formovány do tzv. URL.
- Identifikaci webových serverů podle doménových jmen zajišťují DNS servery.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Pojmy k zapamatování



Internet, WWW, HTTP, (X)HTML, URL, W3C, FTP, DNS

Samostatně

V dalších kapitolách se již začneme věnovat tvorbě statických webových stránek za pomoci jazyka (X)HTML a posléze doplníme styly CSS. V této souvislosti bude potřeba, abyste si nainstalovali a seznámili se se speciálním **textovým editorem PS PAD**.

1. Navštivte stránku <http://www.pspad.com/cz/>
2. Stáhněte a nainstalujte si tento editor <http://www.pspad.com/cz/download.php>
3. Vyzkoušejte si práci v něm
4. V nabídce Soubor, vyberte volbu Nový a z nabízených šablon zvolte HTML.

DŮLEŽITÉ

Srovnejte jazyky HTML a XHTML.

Zde Vám doporučím ke studiu odkaz www.jakpsatweb.cz, případně literaturu uvedenou níže.

- <http://www.jakpsatweb.cz/html/>
- <http://www.jakpsatweb.cz/html/xhtmll.html>

Průvodce studiem



První kapitola nás zavedla do převážně do světa mnoha pojmů a zkratk činnosti služby WWW. Dále se budeme již věnovat praktickým znalostem a práci přímo s technologiemi a jazyky tvorby www stránek.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

2 Základy jazyka HTML

Cíle

Po prostudování této kapitoly budete:

- znát pravidla a syntaxi jazyka XHTML 1.0,
- znát základní značky (tagy) jazyka XHTML 1.0,
- umět vytvořit jednoduchou www stránku za pomoci jazyka XHTML 1.0,
- seznámeni s některými novinkami, které přináší nový standard HTML 5.

Doba potřebná ke studiu

Na prostudování kapitoly Vám bude stačit **150 - 180 minut**.

XHTML

V případě, že jste si na mé doporučení prostudovali vývoj jazyka HTML, tušíte, že nejbližší budoucností je **NOVÁ verze s označením HTML 5**³. Taky jste se ale mohli dozvědět, že tato verze jazyka HTML podléhá ještě stále vývoji a všechny mouchy nejsou ještě plně vychytány. Nemluv o té skutečnosti, že podpora HTML 5 se v hlavních používaných prohlížečích různí.

Tato studijní opora je v příslušných příkladech a rozborech dále věnována **jazyku XHTML 1.0**:

- v současnosti stále nejužívanější jazyk tvorby www stránek,
- vychází z množiny pravidel definovaných XML⁴,
- oproti HTML 4.1 má jednodušší a přehlednější zápis,
- podporuje menší množství značek, je lépe zapamatovatelný,
- lepší přizpůsobitelnost pro SEO (o technologii SEO více později),
- a ještě nějakou dobu se bude úspěšně používat, až jej plně nahradí (X)HTML 5.

Navíc nemusíte mít žádné obavy týkající se zbytečného studia. Novinky, které HTML 5 přináší, v podstatě doplňují to, co už budete umět díky XHTML. Jedná se tedy hlavně o novou funkčnost HTML jako takového. Ale o tom více na konci kapitoly.

³ Chcete-li začnete s HTML 5 třeba zde: http://www.w3schools.com/html/html5_intro.asp

⁴ Standard pro transport a ukládání dat. Více zde: <http://www.w3schools.com/xml/default.asp>

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Základy jazyka XHTML

Příklad 1 – zjednodušený náhled

```
<html>
```

```
<head>
```

```
<title> První příklad </title>
```

```
</head>
```

```
<body>
```

```
<p> Moje<strong>první</strong> www stránka </p>
```

```
</body>
```

```
</html>
```

Hlavička html dokumentu

```
<head> </head>
```

Tělo html dokumentu

```
<body> </body>
```

Nyní si otevřeme program PS PAD. Zvolíme záložku **Soubor** – Nový. V okně **Nový Soubor** zvolíme záložku **Podle šablony**. V seznamu složek s názvy různých standardů rozbalíme XHTML a vybereme šablonu XHTML 1.0 Strict utf-8.

Otevře se dokument **Novy1.xhtml**. Ačkoli budeme používat jazyk XHTML, soubory mají po uložení i tak příponu .html. Mohli bychom psát i .xhtml, ale je to v podstatě jedno.

Dokument Novy1.html obsahuje již některé přednastavené (defaultní) informace. Pokuste se sami:

1. doplnit **text titulku** stránky z Příkladu 1 (text mezi počáteční a koncovou značkou tagu **title**).
2. do **těla stránky** (tj. do místa mezi počáteční a koncovou značkou tagu **body**) okopírujte obsah řádku uvedeného v příkladu 1 (`<p>Mojeprvní....</p>`).

Dokument uložte - následující posloupnost kroků. Klikněte na:

- Soubor
- Uložit jako
- Vyberte místo (nejlépe si na ploše vašeho OS vytvořte složku a pojmenujte ji třeba www) a nazvěte soubor třeba prvni-stranka (v názvu se vyvarujte mezer a zejména diakritiky, pro začátečníka je lépe se vyhnout i kapitálkám). Mezeru nahraďte pomlčkou nebo podtržítkem.
- Názvy html dokumentů jako např. *Moje stránka.html* jsou nepřipustné a v reálném provozu celého webu by způsobovaly nemalé potíže!

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Vytvořili jsme první www stránku, kterou si dvojklikem myši z daného umístění (pokud jste postupovali podle zadání, tak ze složky www na ploše vašeho OS) můžeme otevřít v internetovém prohlížeči. Soubory s příponou .html budou mít ikonu ve tvaru vašeho preferovaného internetového prohlížeče.



TAGY

Základní struktura dokumentu (X)HTML je tvořena TAGY.

Značkám příslušného jazyka, které popisují a formátují text, tabulky, obrázky atp. do kýžené podoby říkáme **TAGY**. Tyto se zapisují do ostrých závorek < > a jsou definovány příslušným jazykem, tj. mají různý význam. Jednoduše řečeno, značkují obsah www stránky a říkají mu, jak se má ve výsledku v internetovém prohlížeči zobrazit.

Každý TAG v jazyce XHTML má svou počáteční a koncovou značku. To je specifikum pro XHTML. Například v našem prvním příkladě si popíšeme TAG – **<p>**, který označuje v těle www stránky text v odstavci.

```
<p>Moje první www stránka</p>
```

Koncová značka daného TAGU se liší pouze přidáním lomítka. Vše co leží mezi počáteční a koncovou značkou daného TAGU je **jeho obsahem** a zobrazí se nám ve výsledku v internetovém prohlížeči.

Logické rozdělení

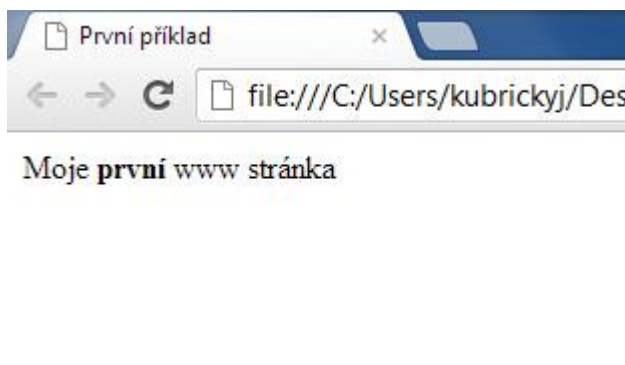
Každá www stránka začíná a končí tagem <html> (viz. příklad 1). Je to tzv. kořenový tag. Každá www stránka obsahuje dvě logické části:

1. **Hlavička dokumentu** – tag <head>. Informace, které jsou uvozeny počáteční a koncovou značkou tagu <head>, jsou určeny pro potřeby internetového prohlížeče a uživatel se s nimi až na dvě výjimky vizuálně nesetká. Uvozují se zde například tzv. **META tagy** o kterých se dozvíte více později⁵. **Důležitou součástí hlavičky je také tag <title>**. Obsah tagu <title> se zobrazí v záhlaví internetového prohlížeče jako titulek stránky.

⁵ Jak vypadají **Metatagy** a jak se zapisují, si můžete prohlédnout ve zdrojovém kódu souboru metatagy.html, který je součástí složky cvičných příkladů kapitoly 2.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický



Obr. 3 Titulek www stránky

2. **Tělo dokumentu** – `<body>`. Vše co je uvozeno počáteční a koncovou značkou tagu `<body>` se zobrazí jako obsah www stránky v internetovém prohlížeči. V našem prvním příkladě to byl vlastně jen jeden text odstavce, Moje **první** www stránka. Pro potřeby zvýraznění slova „první“ jsme použili TAG k tomu určený, TAG ``.

Zásady a konvence jazyka XHTML

- Všechny TAGY se zapisují malými písmeny.
- Všechny TAGY jsou ukončené (mají svou koncovou značku).
- TAGY mohou být zároveň obsahem jiného TAGU (vnořování). Jak jsme např. viděli obsahem tagu `<head>` je tag `<title>`. Nebo obsahem tagu `<body>` je tag `<p>`.
- TAGY se nesmí křížit (tj. nesmí docházet k míchání pořadí koncových značek při vnořování) – viz následující ukázka:

<code><body><p></body></p></code>	... špatně
<code><body><p></p></body></code>	... správně
<code><p></p></code>	... správně

Zápis hlavních TAGŮ www stránek se většinou provádí pro větší přehlednost pod sebe. Jednotlivé TAGY se odsazují tabulátorem od levého okraje tak, aby jejich počáteční a koncová značka měla stejné levé odsazení, a aby tato vzdálenost u jednotlivých TAGŮ rostla s jejich vnořením pod TAGY nadřazené. Podobě jako ve stromovém výpisu adresáře. Nicméně není to povinná zásada a je na vás, jak budete zápis provádět. Já doporučuji uvedený způsob – viz příklad 1.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

K příkladu 1 – odsazení TAGŮ:

```
<html>
  <head>
    <title> Prvni priklad </title>
  </head>

  <body>
    <p> Moje <strong>první</strong> www stranka </p>
  </body>
</html>
```

Hlavním (kořenovým) je TAG **<html>**. Jeho počáteční a koncová značka má nulové odsazení. Tagy **<head>** a **<body>** jsou již tagy vnořené a mají tak určité odsazení. Jelikož jsou oba na logicky stejné úrovni (mají téhož rodiče), mají stejnou vzdálenost od levého okraje. Tagy **<title>** a **<p>** jsou opět tagy vnořené a mají proto opět o něco větší odsazení. Jelikož oba uvozují pouze krátký text, umístili jsme jejich počáteční a koncovou značku na stejný řádek.

Je opravdu prakticky jedno jak si to vyřešíte – **hlavní je, aby se tagy nikdy nekřížily a aby jste se v zápisu poté znovu vyznali**. V dalších příkladech se ale budeme snažit dodržovat uvedenou konvenci.

Pamatujte si:

Fyzické rozmístění textu ve zdrojové části www stránky (ve zdrojovém kódu) nemá žádný vliv na její výslednou vizuální podobu v internetovém prohlížeči. Podobu ovlivňují pouze jednotlivé TAGY a příslušné styly CSS (o nich později).

Čili v internetovém prohlížeči budou úplně stejně zobrazeny následující dva zápisy zdrojového kódu:

```
<p>Moje <strong>první</strong> www stránka</p>
```

```
<p>Moje
  <strong>první</strong>
  www stránka</p>
```

Výsledek bude v obou případech stejný, jako na obrázku 3. Pořád se bude jednat o blokový odstavec.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Příklad 2 – zjednodušený náhled:

```
<html>
<head>
  <meta http-equiv="content-type" content="text/html; charset=utf-8">
  <title>Druhý příklad</title>
</head>

<body>
  <h1>Nadpis</h1>
  <p>Moje druhá www stránka</p>
  <p>Další odstavec.</p>

  <ul>
    <li>Odrážka</li>
    <li>Podrážka</li>
    <li>Urážka</li>
  </ul>
</body>
</html>
```

Tento příklad je stejně jako ten předchozí součástí doplňujících souborů. Nemusíte si jej tedy vytvářet. Stačí soubor *druhy-priklad.html* otevřít v programu PS PAD.

Nás více zajímá obsah **těla** našeho dokumentu. Pomocí tagu **<h1>** jsme vytvořili hlavní nadpis našeho dokumentu. (Pamatujte si, že TAG **<h1>** **smí být v XHTML 1.0 na každé www stránce použit pouze jednou!!!**) Například pro odstavec **<p>** už je to jedno. Těch může být na stránce, jak vidno kolik jen chceme.

Nový je pro nás také tag ****. Ten uvozuje položky **** seznamu.

Podobně jako v MS Word vybíráte jednotlivé nástroje pro to, abyste vytvořili odstavec, seznam, tabulku, barvu písma, tak v jazyce XHTML musíte znát příslušné **tagy** a obsah stránky pomocí něj naformátovat. Je to velmi jednoduché a není vůbec potřeba si všechny tagy pamatovat. Stačí mít po ruce nějakou tu příručku nebo správný edukační web a umět toho využít. Přesto ty nejzákladnější značky jazyka byste měli znát zpaměti. Dále je uveden přehled těch nejpoužívanějších.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Nejpoužívanější tagy

A

`<a>` Definuje odkaz na jiný dokument

B

`<base />` Určuje základní adresu pro relativní odkazy

`<body>` Definuje tělo dokumentu

`
` Zalomí text

C

`<caption>` Definuje titulek tabulky

`<code>` Textový obsah tagu je programový kód

D

`<div>` Vytvoří nový oddíl (blok) dokumentu – blokový element

E

`` Zvýrazní text jako kurzívu

F

`<fieldset>` Vytvoří skupinu prvků (rámeček) ve formuláři

`<form>` Vymezí v dokumentu formulář

H

`<head>` Definuje hlavičku dokumentu

`<hr />` Vloží do dokumentu vodorovnou čáru

`<html>` Vymezí dokument HTML – výchozí TAG

`<h1>` až `<h6>` Vymezí nadpis – můžeme využít 6 úrovní

I

`` Vloží do dokumentu obrázek

`<input />` Vytvoří prvek formuláře (textové pole, seznam,...)

L

`<label>` Definuje popisec ovládacího prvku formuláře

`<legend>` Definuje legendu ovládacích prvků formuláře

`` Vytvoří položku seznamu

`<link />` Připojí jiný dokument (CSS)

M

`<meta />` Definuje doplňující informace o dokumentu

O

`<object>` Vloží do dokumentu objekt



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

 Definuje číslovaný seznam
<option> Definuje položku rozbalovacího seznamu ve formuláři (TAG select)

P

<p> Vymezuje odstavec
<param /> Definuje parametr značky OBJECT
<pre> Zobrazí text jako je zapsaný ve zdrojovém kódu

Q

<q> Označuje citaci

S

<script> Definuje script v dokumentu (např. JavaScript)
<select> Rozbalovací (výběrový) seznam formuláře
 Formátování textu řádkovým elementem span
<strike> Vypíše přeškrtnutý text
 Vypíše tučný text
<style> Definuje styl formátujícího dokumentu
<sub> Zobrazí text jako dolní index
<sup> Zobrazí text jako horní index

T

<table> Vymezení v dokumentu tabulku
<td> Vytvoří buňku tabulky
<textarea> Vytvoří textové pole ve formuláři
<th> Vytvoří buňku v záhlaví tabulky
<title> Vymezení titulek dokumentu
<tr> Definuje řádek buněk v tabulce

U

 Definuje seznam s odrážkami

Entity

Pokud chceme v prohlížeči zobrazit text, prostě ho napíšeme do kódu stránky. **Některé znaky jsou ovšem chápány jako součást jazyka HTML, a tak by se normálně nezobrazily.** Například ostré závorky (znaky „je menší“ a „je větší“) < a > by v textu nešly zobrazit, pže internetový prohlížeč je prostě chápe jako součást jazyka HTML. Pro tento účel vznikly entity, které se vzápětí rozšířily o mnoho dalších znaků.

Znakové entity se zapisují jako sekvence znaků. Sekvence začíná znakem & a končí středníkem. Mezi nimi je krátký písmenkový zápis daného znaku.

Asi nejpoužívanější entitou je pevná mezera. Představuje jediný způsob, jak na stránce zobrazit několik mezer po sobě (odhlédněme nyní od faktu, že dvě mezery za sebou jsou v typografii chybou).



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Zapíše se jako ** **; (z angl. non-breaking space).

Příklady Entit

| Znak | Entita | Znak | Entita |
|--------------|---------|------|----------|
| pevná mezera | | ± | ± |
| < | < | × | × |
| > | > | ® | ® |
| & | & | © | © |
| " | " | € | € |
| » | » | « | « |

Obrázky a hypertextové odkazy

Příklad 3 – zjednodušený náhled:

```
<html>
<head>
  <meta http-equiv="content-type" content="text/html; charset=utf-8">
  <title>Třetí příklad</title>
</head>

<body>
  <h1>Nadpis</h1>
  <p>Moje druhá www stránka</p>
  <p>Další odstavec.</p>
  <ul>
    <li>Odrážka</li>
    <li>Podrážka</li>
    <li>Urážka</li>
  </ul>

  <p><a href="http://www.seznam.cz" title="Seznam">Seznam</a></p>
  <p><a href="http://www.atlas.cz" title="Atlas">Atlas</a></p>

  <p><a href="obrazky/foto.jpg" title="Foto">Ukaž foto</a></p>

</body>
</html>
```

Máme před sebou třetí příklad této kapitoly. Opět jej máte k dispozici s příklady k druhé kapitole. Otevřete si jej v PS PADU a prohlédněte si jeho zdrojový kód spolu s komentářem.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Oproti druhému příkladu se tělo dokumentu zase o něco rozrostlo. Pod seznamem nám přibyl tag ``. Tag `img` je jako jeden z mála v jazyce XHTML nepárový, tj. jeho koncová značka se nezapisuje, **ale nahrazuje se lomítkem na konci**.

```

```

Tag `` umožňuje vložit do www stránky obrázek. Povinná struktura tagu `` je následující:

Nejdůležitější je **atribut** (pro nás nyní nový výraz) **src**. Do něj se pomocí rovnítka a uvozovek zapisuje cesta ke zdrojovému obrázku, který se na naší stránce zobrazí. Pomocí **relativní cesty** zapíšeme adresu obrázku.

Relativní cesta (odkaz) znamená, že se neodkazujeme přímo na absolutní umístění obrázku jako např.:

```
C:\Documents\admin\Plocha\www\obrazky\kvetina.jpg
```

My se odkazujeme **relativně**. Tzn., že prohlížeč vyhodnotí umístění naší www stránky a v podřízené složce **obrazky** hledá soubor *kvetina.jpg*. Pozor, jak vidíte, je potřeba zapisovat cestu k obrázku i s danou příponou.

Na webu se nejčastěji používají obrázky ve formátu **JPEG, GIF** nebo **PNG**. **Rovněž si dávejte pozor na zápis příjmeny souboru** – na webovém serveru **NENÍ** totéž *kvetina.jpg* a *kvetina.JPG*.

Obr. 4 Obrázky a odkazy na www

Druhým atributem tagu `` je atribut **alt**, který popisuje daný obrázek a zároveň prohlížeči nabízí alternativní zobrazení textu místo např. chybějícího obrázku.

Pozor: I v případě, že obrázek je k dispozici a prohlížeč jej zobrazí, atribut **alt nikdy nevynechávejte**. O jeho důležitosti se přesvědčíte v kapitole věnované problematice **SEO**.

Další nové prvky v našem třetím příkladu jsou **hypertextové odkazy**, které jsou nepostradatelnou součástí www stránek. Umožňují nám pouhým kliknutím myši přenést se na cíl odkazu.



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

`Seznam`

Hypertextový odkaz se uvozuje do párového tagu `<a>`. Má opět povinné atributy, které je potřeba přesně dodržet. Tím nejdůležitějším je podobně jako u obrázku cíl odkazu. Zapisuje se zde pomocí atributu **href**. Opět pomocí rovnítka a uvozovek **nastavíme** cíl odkazu. Např. portál seznam. Hypertextový odkaz mívá ještě doporučován atribut **title**, což je text, který se uživateli zobrazí v bublině při najetí kurzorem na odkaz.

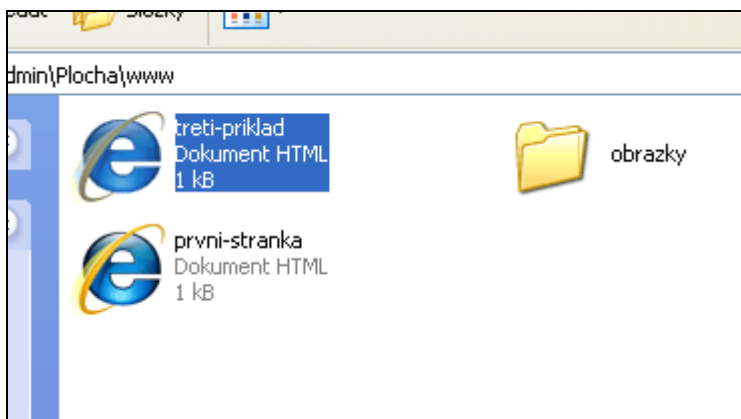
Text, který se v prohlížeči jako odkaz zobrazí je uvozen mezi počáteční a koncovou značku tagu `<a>`.

Stejně jako v případě cesty k obrázku se i u hyper. odkazů mohou použít relativní i absolutní cesty. Můžeme se odkazovat na jinou www stránku, třeba příklad 1, nebo také na fotografii uloženou ve složce **obrazky**.

`Ukaž foto`

V případě, že byste se chtěli odkazovat na stránku prvního příkladu (první-priklad.html) a máte ji stejně jako třetí příklad uloženou ve složce kapitola 2, vypadal by odkaz následovně:

`Moje prvotina`



Obr. 5 Relativní odkazování

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Multimediální objekty

Přehrávání videa a zvuku představuje v XHTML 1.0 docela problém. Závisí na tom, jaké soubory chcete přehrávat, jakou metodu (tagy HTML) pro to použít a zdali má uživatel k dispozici ten či onen internetový prohlížeč a k tomu ještě nainstalován správný plugin.⁶ Tyto problémy se snaží řešit verze jazyka HTML5 a XHTML5.



Zatím se pro „téměř“ spolehlivou projekci videa požívá přehrávač **Adobe Flash**, nicméně to je kapitola sama o sobě a pro začátečníka celkem náročná. Proto bývá nejmoudřejší využít služby třetí strany. Pro účely prezentace videa na vlastních www stránkách je dobré využít například serveru **YouTube**.

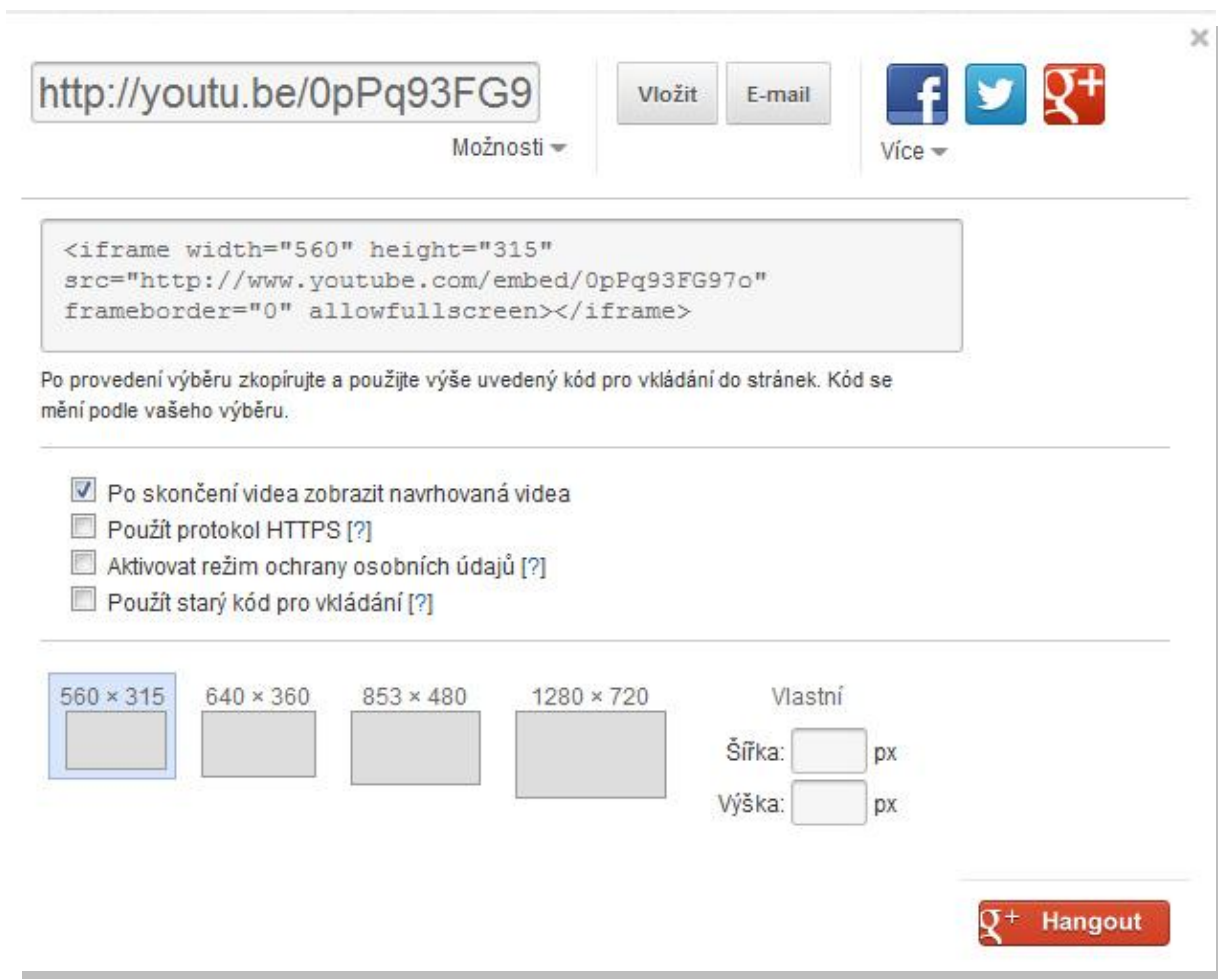
Vybereme video nebo jej přímo uploadujeme na server YouTube. Poté stačí okopírovat poskytnutý kód a vložit do zdrojového kódu naší www stránky a je hotovo.

⁶ Pokud byste rádi vyzkoušeli i další možnosti navštivte: http://www.w3schools.com/html/html_videos.asp

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický



Obr. 6 Zdrojový kód z YouTube

Nápověda PS PADu a ONLINE zdroje

Zapamatovat si všechny tagy v případě, že nepíšeme www stránky každý den, je obtížný úkol. **V programu PS PAD je vložena implicitní nápověda.** Klepněte na libovolné místo v dokumentu a dále stačí kombinace kláves levý CTRL a mezerník – vyzkoušejte.

Ideální on-line zdroje:

Český: <http://www.jakpsatweb.cz/html/rejstrik.html>

Anglický: <http://www.w3schools.com/html/default.asp>

V anglickém manuálu můžete rovnou zkusit jednotlivé tagy. Jedná se o nejlepší a nejširší online zdroj vůbec.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Co přináší HTML5



Dříve než si představíme novinky, které s sebou přináší HTML5, řekněme si pár známých skutečností:

1. HTML5 je stále ještě připravovaným standardem HTML.
2. HTML 5 navazuje nejen na HTML 4, ale také XHTML a DOM Level 2.
3. HTML5 je navrženo tak, aby obsahovalo vše potřebné bez nutnosti dalších pluginů (např. Flash playeru). Od animací k aplikacím, hudbě k filmům, a může být také použito k vytvoření složitějších aplikací, které běží přímo v prohlížeči v offline režimu.
4. HTML5 je také multiplatformní (je jedno, jestli používáte tablet nebo smartphone, netbook, notebook, Smart TV či PC).
5. HTML 5 tvoří pracovní skupiny patřící pod společnosti AOL, Apple, Google, IBM, Microsoft, Mozilla, Nokia, Opera, a stovky dalších dodavatelů.
6. HTML5 je stále nedokončený. Nicméně všechny hlavní prohlížeče podporují mnoho nových prvků HTML5 a API.

Jak vidno HTML 5 si klade poměrně komplexní a užitečné cíle a snaží se situaci kolem tvorby www stránek především **zjednodušit**, odtrhnout od spousty doplňujících technologií a přinést mnohé přelomové a inovativní možnosti.

Bohužel detailnější popis HTML 5 by vydal na další samostatnou studijní oporu, a proto se na hlavní novou funkčnost podíváme pouze ve zkratce.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Minimaliazce

Deklarace typu dokumentu:

XHTML 1.0 Strict

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

HTML 5

```
<!DOCTYPE html>
```

Kódování dokumentu:

XHTML 1.0 Strict

```
<meta http-equiv="content-type" content="text/html; charset=utf-8" />
```

HTML 5

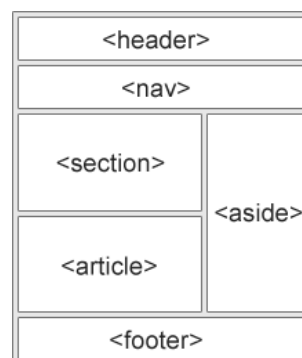
```
<meta charset="utf-8">
```

Popisnější zdrojový kód

Z hlediska rozdělení obsahu www stránky do bloků se v XHTML používá tag **<div>**. Jednotlivé divy se od sebe musí odlišit pomocí tříd a identifikátorů a tím se zdrojový kód www stránky může komplikovat.

HTML5 přináší nové sémantické prvky, které jasně definují různé části webové stránky:

- <header>
- <nav>
- <section>
- <article>
- <aside>
- <figcaption>
- <figure>
- <footer>



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Multimédia s menší závislostí na zásuvných modulech

Pro video, audio nebo vektorovou grafiku už nepotřebujeme Flash nebo SilverLight. Pro audio se používá velmi jednoduchý TAG **<audio>**. V závislosti na typu internetového prohlížeče, lze přehrávat formáty zvuku **mp3**, **wav** a **ogg**.

```
<audio controls>
  <source src="horse.ogg" type="audio/ogg">
  <source src="horse.mp3" type="audio/mpeg">

  Váš prohlížeč nepodporuje element AUDIO.
</audio>
```

V případě videa je nový TAG **<video>**. V závislosti na typu internetového prohlížeče, lze přehrávat formáty zvuku **mp4** a **ogg**.


```
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">

  Váš prohlížeč nepodporuje element VIDEO.
</video>
```

Element CANVAS

HTML5 element **<canvas>** se používá ke kreslení grafiky za běhu, pomocí skriptování (obvykle JavaScript). Element **<canvas>** je jen kontejner pro grafiku. Musíme použít skript, který grafiku vykreslí. Existuje několik metod pro kreslení cest (čar), kvádrů, kruhů, textu a přidávání obrázků.

```
<!DOCTYPE html>
<html>
<body>
  <canvas id="myCanvas" width="200" height="100" style="border:1px
solid #c3c3c3;">
    Your browser does not support the HTML5 canvas tag.
  </canvas>
  <script>
    var c=document.getElementById("myCanvas");
    var ctx=c.getContext("2d");
    ctx.fillStyle="#FF0000";
    ctx.fillRect(0,0,150,75);
  </script>
```



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

```
</body>
```

```
</html>
```

Lepší formuláře

HTML5 přináší nové ovládací prvky uživatelského rozhraní formulářů. Bohužel internetové prohlížeče je implementují postupně, tzn., že si na plnou podporu těchto novinek budeme muset ještě chvíli počkat. Z těch zásadních se jedná o:

Input Type: date

Formulářové políčko, umožňující vybrat konkrétní datum z grafického kalendáře.

Input Type: email

Formulářové textové políčko, které umí automaticky zkontrolovat validitu zadaného emailu.

Input Type: range (posuvník)

Formulářové pole umožňující pomocí posuvníku nastavit hodnotu z daného intervalu.

A další...

Pro zájemce

HTML5 toho přináší více. Webové sockety, Úložiště na straně klienta, Podporu CSS3, Vizuální efekty, Techniku Drag and Drop nebo také Geolokaci uživatele. Díky svým vlastnostem, kombinaci s CSS a JavaScriptem lze např. vytvářet i jednoduché hry.⁷ Pro zájemce doporučuji začít na oficiálním výukovém webu W3Schools.com.

Shrnutí

- Značkám jazyka XHTML říkáme **tagy**. Každý **tag** má svou počáteční a koncovou značku.
- Koncové značky doplňujeme před názvem tagu lomítkem. Pouze v případě nepárových tagů, jako např. ``, `
` nebo `<input />` píšeme lomítko na konec, přímo do počátečního tagu.
- Každá www stránka je uvozena mezi počáteční a koncovou značku kořenového tagu `<html></html>`.
- Logická struktura www stránky obsahuje 2 části: **hlavičku** a **tělo**.

⁷ Více v publikaci **Programujeme hry v HTML 5** od Makzana. 2013.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

- XHTML tagy se nesmí křížit, píšou se malými písmeny a vše co je v nich uvozeno nazýváme **obsahem** tagu.
- Na interní dokumenty a obrázky se můžeme odkazovat **relativním odkazem**.
- Součástí tagu jsou tzv. **atributy**. Více o nich v příští kapitole.
- Na webu se používají zejména obrázky ve formátu JPEG, GIF, PNG.
- Nový standard HTML je označen HTML5. Přináší zásadní novinky týkající se grafiky a multimédií, vylepšení formulářů, možnost offline aplikací, aj.

Pojmy k zapamatování



XHTML, Tagy, Atributy, Logická struktura www stránky, HEAD, BODY, HTML5

Průvodce studiem



Naše první pokusy ještě nejsou žádným zázrakem, ale příště již v kombinaci s jazykem **CSS** začneme vytvářet mnohem komplexnější a graficky vzhlednější www stránky.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

3 Kaskádové styly

Cíle

Po prostudování této kapitoly budete:

- znát pravidla a syntaxi jazyka CSS,
- umět vytvořit a upravit jednoduchý layout webu pomocí XHTML a CSS,
- seznámeni s některými novinkami CSS 3.

Doba potřebná ke studiu

Na prostudování kapitoly Vám bude stačit **120 minut**.

Značkovací jazyk (X)HTML vytvoří doslova „kostru“ webových stránek. Jejich výslednou podobu (pakliže se nespokojíme pouze s klasickým bílým pozadím, černým textem atd.) jim ale dají tzv. **kaskádové styly CSS**. Jejich použití je velmi jednoduché a nevyžaduje stejně jako HTML žádné zvláštní programátorské dovednosti.



CSS

Kaskádové styly (CSS) jsou jazykem pro popis způsobu zobrazení stránek napsaných v jazycích HTML, XHTML nebo XML. Jednoduše řečeno, CSS nám umožňuje naformátovat základní šablonu napsanou (v našem případě) v jazyce XHTML do graficky efektivnější podoby.

V podstatě tak můžeme říci, že díky CSS jsme schopni na webových stránkách pracovat efektivně s grafikou.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Součástí výkladu této kapitoly jsou opět příklady, konkrétně složka kapitola 3. Příklady obsahují důležité komentáře, kterým byste měli věnovat zvýšenou pozornost.

Srovnejte:

Otevřete složku **CSS-cviceni 1**. Obsahuje 3 soubory. Vyzkoušejte:

1. Nejprve si v internetovém prohlížeči zobrazte soubor *stranka-bez-css.html*.
2. Poté si v internetovém prohlížeči zobrazte soubor *stranka-s-css.html*.

Jedná se o dva prakticky totožné dokumenty vytvořené za pomoci jazyka XHTML, ale *stranka-bez-css.html* neobsahuje formátování vzhledu za pomoci CSS, kdežto v druhém případě již ano. Sami vidíte, jak graficky velmi užitečnou práci umožní právě kaskádové styly CSS.

Vše má na svědomí třetí soubor ve složce *CSS-cviceni 1 style.css* a TAG `<link />` vložený v hlavičce dokumentu *stranka-s-css.html*. V souboru *style.css* jsou nadefinovány příslušné styly, které nám umožňují měnit vzhled prakticky všech prvků na naší stránce.



Nadpis první úrovně by neměl chybět

Nadpis druhé úrovně má taktéž velkou váhu

Text v odstavci je na www stránkách neodmyslitelnou součástí. Je vhodné taktéž používat tučně zvýrazněné **klíčové pojmy** a kurzívou zvýrazněné *citace*.

Všimněte si v souboru *style.css* jak se zapisují barvy. Existuje 16 základních bezpečných barev, které lze zapisovat jejich ang. názvy. [Základní barvy](#).

Všechny barvy lze ale zapsat pomocí # a hexadecimálního čísla. Pokud nemáte třeba Photoshop, tak hexadecimální kód barev můžete získat třeba zde. [On-line generátor](#).

[Nevypadalo by to s CSS lip???](#)

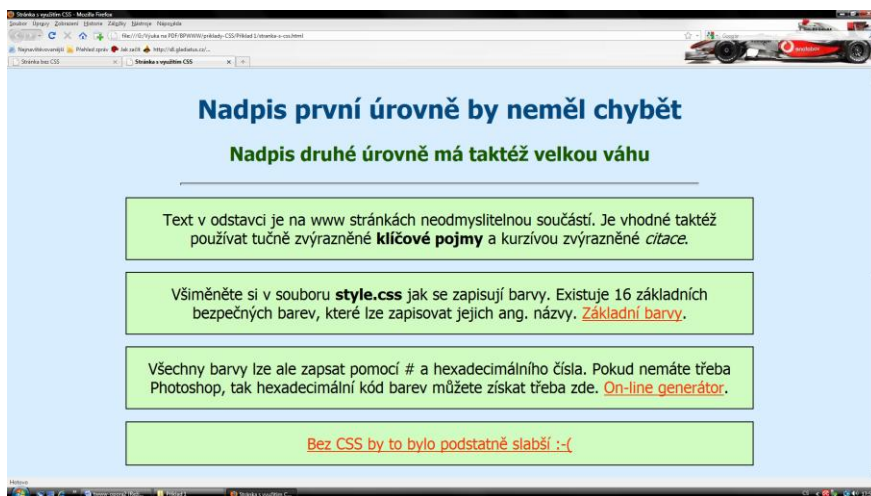


Obr. 7 Stránka bez využití CSS

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický



Obr. 8 Stránka s využitím CSS

Poznámka: Pečlivě si přečtěte obsah (prostudujte i zdrojový kód) souborů a navštivte externí odkazy, které jsou v příkladech zveřejněny. Budou se vám při vytváření vašich www stránek jistě hodit.

V dalším kroku porovnáme zdrojový kód obou uvedených stránek.

- V případě stránky **stranka-bez-css.html** se jedná o klasickou XHTML stránku.
- Ovšem **stranka-s-css.html** obsahuje již některé nutné doplňky, které si nyní vysvětlíme:

Součástí hlavičky `<head></head>` je v této chvíli nejdůležitější odkaz na externí soubor, kde se nadefinované styly CSS nacházejí:

```
<link href="style.css" type="text/css" rel="stylesheet" />
```

V našem příkladu soubor **style.css** leží ve stejné složce jako dokument **stranka-s-css.html**, takže se pomocí atributu **href** odkazujeme přímo na daný soubor. Atribut **type** vymezuje o jaký soubor se jedná a **rel** uvozuje popis daného dokumentu.

Jelikož je TAG `<link />` tagem nepárovým, nesmíme zapomenout na koncové lomítko, abychom dodrželi pravidla jazyka XHTML!

Tři způsoby zápisu CSS

Existují **3 způsoby**, jak vložit do dokumentu XHTML styly CSS. První dva způsoby umožňují zápis přímo do dokumentu XHTML (nepraktické, nepřehledné a neefektivní), třetím se v dokumentu XHTML



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

pouze odkazujeme na externí soubor (ideální, protože tím oddělíme obsah – *html dokument* od formátování – *soubor se styly CSS*).

1. Přímý inline zápis stylu pomocí atributu **style** (styly jsou přímou součástí počátečních značek jednotlivých HTML tagů – jak vidíte níže, dochází tak k nežádoucímu míchání obsahu a jeho formátování):

Ukázka:

```
<p style="color: red; text-decoration: underline">Tento odstavec bude  
červený a podtržený.</p>
```

2. Zápis stylů do elementu **style** (styly jsou opět přímou součástí html dokumentu, zapisují se jako přímá součást hlavičky www stránky):

Ukázka:

```
<style type="text/css">  
  p {  
    color: red;  
    text-decoration: underline;  
  }  
</style>
```

3. Připojení externího souboru **pomocí tagu <link />**.

Třetí způsob je pro své nesporné výhody nejvyužívanější. Uvedeme ty nejzásadnější plusy:

- Oddělení obsahu dokumentu od jeho formátování.
- Možnost využít jeden soubor CSS pro více dokumentů (odpadá duplicita CSS stylů, není potřeba styly na každé stránce zapisovat znovu).
- Přehlednější zápis pro vývojáře.
- Rychlejší zobrazení v internetovém prohlížeči (soubor je po dobu brouzdání po stránkách uložen v operační paměti klienta).

Dále proto budeme styly CSS umísťovat vždy do externího souboru (přípona .css).

Poznámka: Všimněte si rozdílu pro označení komentářů v dokumentu **HTML** (`<!--text komentáře-->`), kdežto v **CSS** (`/* text komentáře */`).

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

CSS – syntaxe

Jazyk CSS sestává z tzv. **pravidel**. Každé pravidlo obsahuje **selektor** a **blok deklarácí**. Každý blok deklarácí pak obsahuje **seznam deklarácí** a každá deklaráce sestává z **vlastnosti**, následuje dvojtečka : a **hodnota vlastnosti**. Každá deklaráce končí středníkem ;

Příklad pravidla:

```
body {  
    background-color: white;  
    color: black;  
    margin: 10px;  
}
```

Celý výše uvedený blok je pravidlo, „**body**“ je selektor, to v závorkách blok deklarácí. Řádek „**background-color: white;**“ je deklaráce samotná, „**background-color**“ je vlastnost a „**white**“ je hodnota této vlastnosti.

Celý kód v uvedeném příkladu je vztažen k elementu těla www stránky <body>, což znamená, že vlastnosti tohoto pravidla ovlivní vzhled celé www stránky.

- Nastavuje barvu pozadí stránky na bílou (background-color: white;),
- Barvu veškerého textu na černou (color: black;)
- a vnější okraje celé stránky na hodnotu 10 pixelů (margin: 10px;).

Stačí být trochu obeznámen se základy angličtiny a význam je jasný.

Selektory

CSS definuje mnoho různých selektorů, které obvykle můžeme kombinovat. Základní selektory mají stejný název jako příslušné HTML TAGY (např. body, h1, p, atd). Uvedme:

body – Deklarace pro selektor **body** se vztahují na formátování celého těla www stránky.

body p – Tyto deklaráce budou platit pro všechny elementy p (resp. odstavce), které se nachází v elementu **body** (uvnitř těla www stránky), v jakékoliv hloubce.

.trida – Tyto deklaráce budou platit pro všechny elementy, které mají v HTML nastavenou třídu **trida**.

To se provádí pomocí HTML atributu **class**. V dokumentu HTML se pak odkážeme například takto:

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

<p class="trida">Nastylovaný odstavec</p>

#main – Tyto deklarace budou platit pro všechny elementy, které mají v HTML nastavený identifikátor id. To se provádí pomocí HTML atributu id. V dokumentu HTML se pak odkážeme například takto:

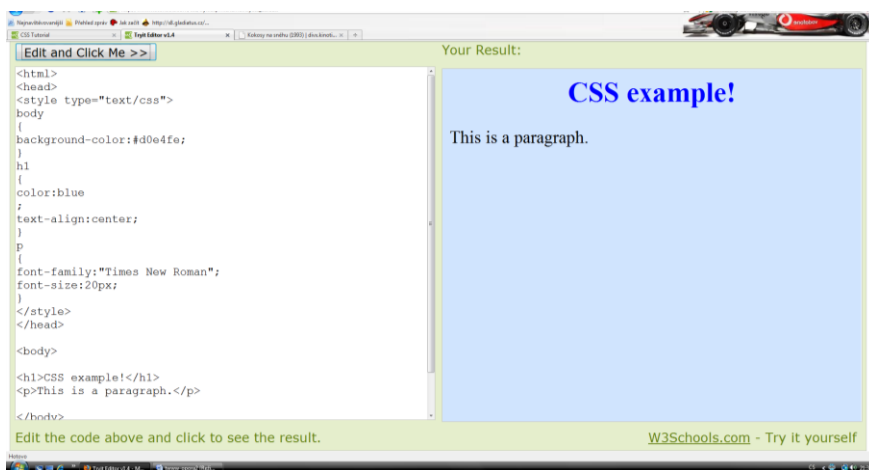
<h2 id="main">Hlavní nadpis</h2>

Třídy a Identifikátory byly původně zavedeny pro to, aby bylo možné vytvářet více verzí stylů pro jednotlivé elementy. Jinak bude graficky formátován obyčejný odstavec <p> a jinak může vypadat odstavec s třídou – např. <p class="trida"></p>.

Více se na toto téma zde rozepisovat nebudu, protože vše lépe pochopíte až v praktických příkladech. Zatím si dobře prohlédněte soubor **style.css** v příkladu 1. Zdrojový kód a jednotlivé vlastnosti CSS jsou tam okomentovány a vysvětleny.

On-line tester

Na webu W3Schools si můžete vyzkoušet, jak se pracuje se styly CSS a rovnou si prohlížet výsledky změn které provádíte. Ve zkušebníce je využit druhý způsob vložení stylu do dokumentu, ovšem to nám jistě nebrání v testování:



Obr. 9 CSS Zkoušečka

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

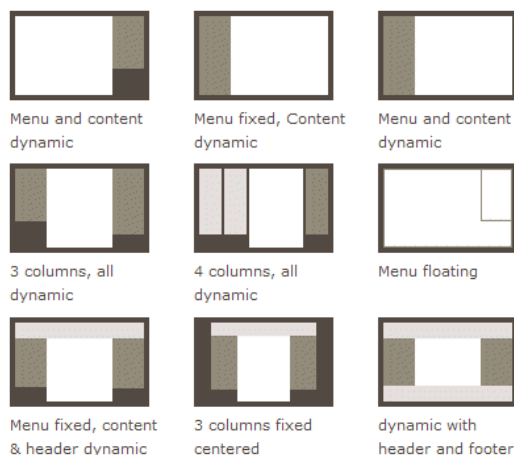
Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Layout www stránek

Dlouhou dobu se řešilo, jakým způsobem rozložit celý obsah webu do samostatných kompaktních bloků. Chtěli jsme mít na svých stránkách třeba horní blok jako záhlaví celé stránky – např. s logem společnosti, názvem školy atp. Poté levý sloupec pro menu stránek s hypertextovými odkazy a pak větší blok pro obsah jednotlivých stránek. Dříve se k těmto účelům používaly **Rámy**, tzv. **frames**. (více o nich např. zde <http://www.jakpsatweb.cz/html/ramy.html>).

Rámy jsou ale z více dnes závažných důvodů již zavržené řešení a byly nahrazeny tzv. tabulkovým layoutem. Obsah webu byl celý uvozen do jedné velké tabulky a bylo po starostech. Nicméně i toto řešení není z nejmoudřejších a proto se dnes vytváří tzv. **CSS layout**.



Obr. 10 CSS Layout

(zdroj: Coding Horror. Lets Build Grid. <http://blog.codinghorror.com/lets-build-a-grid/>)

Pomocí **layoutu** si můžeme webovou stránku rozdělit do oblastí, do kterých pak vkládáme dílčí části našich stránek. Na internetu najdete dokonce spoustu on-line generátorů, které dle vašeho zadání vytvoří potřebný styl CSS. Zkuste sami zapátrat, určitě nějaký najdete⁸.

My si nyní takový layout vytvoříme sami. Máme jeden připravený ve složce **CSS-cviceni 2**. Otevřeme si oba soubory v programu **PS PAD**. Soubor *index.html* obsahuje v hlavičce nezbytný, nám již známý odkaz na soubor se styly.

```
<link href="style.css" type="text/css" rel="stylesheet" />
```

⁸ Např: <http://www.cssportal.com/layout-generator/>



evropský
sociální
fond v ČR



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Tělo dokumentu obsahuje TAGY, které možná ještě tak dobře neznáte, nicméně jsou základním prvkem všech www stránek tvořených za pomoci jazyků XHTML a CSS. Jsou to **blokové tagy <div>**.⁹ **Tyto tagy označují na www stránce bloky**. Může se jednat o celé velké bloky layoutu nebo i některé dílčí, jako např. bloky textu, obrázků, atp.

Pokud jste si již prohlédli soubor *index.html* v internetovém prohlížeči, určitě jste postřehli, že náš layout se skládá ze tří viditelných částí (bloků). Tyto bloky definujeme právě za pomoci tagu **<div>**. V příkladu jsou celkem tři bloky.

Samotná www stránka má bílé pozadí, je široká 1000px a je umístěna na střed obrazovky. Tak ji formátuje následující kód ze souboru *style.css*. Formátování na střed zajišťuje automatický levý a pravý vnější okraj.

```
body, html {  
  
    background: white;  
  
    width: 1000px;  
  
    margin: 0;  
  
    margin-left: auto;  
  
    margin-right: auto;  
  
}
```

Úkol:

Pokuste se sami nahradit bílé pozadí stránky za barvu stříbrnou. Stejně jako bílou barvu, lze i barvu stříbrnou zapsat do souboru CSS přímo jejím anglickým názvem.

Celý layout má šířku 1000px (obrazových bodů) a je umístěn relativně, na střed. To zajistí hlavní (pomocný) div s identifikátorem center. Všechny ostatní **divy** jsou umístěny (vnořeny) do divu center.

V souboru **index.html** vypadá tělo dokumentu následovně:

```
<div id="zahlaví">  
</div>  
  
<div id="levy_sloupec">
```

⁹ Tagy **<div>** se HTML 5 snaží eliminovat. Více v předchozí kapitole, podčást HTML 5 – Popisnější zdrojový kód. Ovšem při dobrém návrhu je lze stále úspěšně používat. Záleží na zvyku a na vás, pro kterou variantu se rozhodnete nebo jakou již připravenou šablonu budete využívat – více v následující kapitole.



evropský
sociální
fond v ČR



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

```
</div>

<div id="pravy_sloupec">
</div>
```

ID je tzv. identifikátor, který se v obsahu **www stránky může vyskytovat pouze jednou**. Tj. ve zdrojovém kódu naší www stránky se nemůže na více místech objevit třeba **<div id="levy_sloupec">**. U tvorby layoutu je to logické, protože více divů s identifikátorem *levy_sloupec* nebo *zahlaví* již nebudeme využívat.

V případě, že byste chtěli využívat jeden nastýlovaný **div** vícekrát, užila by se místo identifikátoru **třída**. Ta se více využívá u textových elementů. Například odstavce.

```
p.zarovnany {
    text-align: justify;
}
```

V dokumentu HTML to pak vypadá následovně:

```
<p class="zarovnany"> Nějaký text </p>
```

A na jiném místě dokumentu bychom mohli opět použít předešlou definici. A to kolikrát jen budeme potřebovat.

Další bloky, tj. **zahlaví**, **levy_sloupec**, **pravy_sloupec** mají následující formátování:

```
div#zahlaví {
    background: blue;
    width: 1000px;
    height: 200px;
    margin: 0;
    float: left;
}

div#levy_sloupec {
    background: red;
    width: 300px;
    height: 500px;
    margin: 0;
    float: left;
}

div#pravy_sloupec {
    background: green;
```




evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

```
width: 700px;  
height: 500px;  
margin: 0;  
float: left;
```

```
}
```

Všimněte si zejména vlastnosti **float**. Jedná se o obtékání stylovaného elementu vzhledem k ostatním elementům stránky. To znamená, že například pravý sloupec se umístí zleva vedle levého sloupce (bude ho obtékat zleva). Jelikož **součet jejich šířky** 300 + 700 px je roven celkové šířce 1000px a protože oba, jak levý_sloupec tak pravý_sloupec mají obtékání nastaveno zleva – **float: left;**

Levý sloupec již blok záhlaví zleva obtékat nemůže, protože samotné záhlaví již má max. šířku 1000px (maximální neboli shodnou vzhledem k nastavené šířce těla www stránky) a proto se levý sloupec umístí hezky pod záhlaví, zarovnan vlevo. Ostatní vlastnosti jako *šířka (width)* či *výška (height)* jsou vám již určitě jasné.

CSS3

Stejně jako HTML je i CSS jazykem, který prochází stálým vývojem. Posledním resp. nejnovějším standardem je verze označená **CSS3**. A nutno říci, že stejně jako HTML5 přináší mnohá vylepšení a navíc je zpětně kompatibilní s předchozími verzemi. A stejně jako HTML5 **prochází stálým vývojem** a integrací do nejnovějších verzí používaných internetových prohlížečů.

Jako zásadní novinky byly přidány ty funkčnosti, které dříve vyžadovaly více grafiky, složitější návrh vlastních CSS stylů nebo nutnost řešit některé návrhy v kombinaci s využitím skriptů programovacího jazyka.

Web W3Schools.com uvádí jako nejdůležitější:

- Přidání nových selektorů
- Vylepšená práce s pozadím a rámečky
- Vylepšená práce s obrázky a nahraditelný obsah
- Textové efekty
- 2D/3D Transformaci
- Animace
- Popora vícenásobného sloupcového layoutu
- Uživatelské rozhraní

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Není předmětem tohoto studijního materiálu detailně popisovat jednotlivé možnosti CSS 3. Princip kaskádových stylů obecně by vám nyní již měl být zřejmý, a pokud se zajímáte o některé výše zmíněné body, navštivte stránku W3Schools.com a dozvíte se více.¹⁰

Shrnutí kapitoly

- Kaskádové styly CSS slouží pro formátování vzhledu www stránek.
- Existují tři způsoby zápisu stylů CSS.
- Nejvhodnější způsob zápisu a připojení stylů CSS k dokumentu HTML je prostřednictvím externího souboru s příponou .css.
- V dokumentu HTML se externí soubor CSS připojuje v hlavičce pomocí TAGU `<link />`.
- Styly CSS se zapisují do pravidel, které se skládají ze Selektoru, Bloku deklarací, Vlastností a jim přidělených hodnot.
- Selektory CSS jsou deklarovány jako názvy elementů HTML, které stylují. Lze využívat rovněž **tříd** a **identifikátorů**. Třidu lze v HTML dokumentu použít neomezeně, identifikátor pouze jednou.
- Layout webu je fyzické rozložení elementů www stránky tvořený kombinací HTML a CSS.
- Blokované elementy Layoutu webu tvoří TAGY `<div>`. V layoutu jsou jednotlivé divy rozlišeny pomocí identifikátorů a tříd.
- Nejnovější standard kaskádových stylů je označen CSS3.

Pojmy k zapamatování



CSS, kaskádové styly, TAG `<link />`, pravidlo, selektor, deklarace, vlastnost a hodnota, třída, identifikátor, layout, TAGy `<div>`, CSS3

¹⁰ http://www.w3schools.com/css/css3_intro.asp

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Samostatně

Sami vytvořte následující CSS layout který:

- bude umístěn na střed
- bude mít šířku 1100px
- bude mít záhlaví o šířce 1100px, a výšce 150px
- bude mít 3 sloupce o volitelné šířce (celkový součet 1100px) a výšce 500px
- bude mít patičku o šířce 1100px a výšce 80px

Použijte libovolné barvy na pozadí. WWW stránku nazvěte **index.html** a soubor CSS **style.css**. Layout by měl vypadat asi nějak takto:



Můžete použít šablonu, která je přiložena v CSS-cvičení 2. Stačí ji jen poupravit. K prostudování a jako zdroj dalších informací vám k CSS doporučuji následující:

Řešení úkolu najdete ve složce ukol-layout.

Více ke kaskádovým stylům na Jak Psát Web: <http://www.jakpsatweb.cz/css/>

A W3Schools.com: <http://www.w3schools.com/css/>

Průvodce studiem



V této chvíli známe vše potřebné pro to, abychom mohli tvořit šablony www stránek, které už budou vypadat tak nějak k světu 😊. Ale cesta je to stále trnitá a pro začátečníka pořád ještě náročná. Ale proč stahovat kalhoty, když brod je ještě daleko. Existuje sposta cest, jak si tvorbu www stránek ulehčit a přitom nic neubrat na kvalitě.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

4 HTML a CSS

Cíle

Po prostudování této kapitoly budete:

- umět vytvořit jednoduchou šablonu www stránky za využití XHTML a CSS,
- vyhledat vhodnou free šablonu www stránek pro využití ke konkrétním účelům,
- umět upravit free šablonu.

Doba potřebná ke studiu

Na prostudování kapitoly Vám bude stačit **120 minut**.

Šablony

Z **třetí kapitoly** již dobře víme, že Kaskádové styly (CSS) lze velmi dobře využít pro grafickou úpravu www stránek. Taktéž jsme poznali, že pomocí externího zápisu do souboru je možno tyto styly úplně oddělit od obsahu www stránky.

Dále jsme poznali, že styly se dnes využívají pro tvorbu **layoutu** webové stránky. Na internetu lze dokonce nalézt spoustu dostupných šablon ke stažení, které můžeme pro své webové stránky využívat.

Není to ale všechno. S kaskádovými styly CSS lze provádět dokonalá kouzla a je spousta nadšenců, kteří připravují kompletní šablony pro běžné uživatele, a pro ty, kteří se s tvorbou základního návrhu nemohou nebo prostě jen nechtějí zdržovat. Na příložených odkazech si můžete prohlédnout celé sady free šablon a už jen vybrat tu, která vám bude vyhovovat.¹¹

Grafika

Při tvorbě kvalitních a graficky poutavých stránek si dnes nevystačíme bez grafického editoru. Nejvyužívanějším je bezesporu **Photoshop** nebo free **GIMP**, ale úspěch slaví i vektorový **Illustrator** nebo **CorelDraw**. Prakticky všechny šablony se nejprve „kreslí“ v těchto programech a pak se pomocí CSS a HTML převádějí do podoby dokumentů webu. Pakliže nemáte žádný grafický program k dispozici a nebo nejste zdatným grafikem, volné šablony jsou pro vás jako stvořené. Pokud budete

¹¹ <http://www.freemtemplatesonline.com/templates/>
<http://www.freehtml5templates.com/>

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

trpěliví, jistě na webu najdete šablonu vhodnou právě pro vaše záměry. Rozhodně vás od této možnosti neodrazují, ba naopak. Se znalostmi (X)HTML a CSS si můžete danou šablonu upravit pro své potřeby a můžete tak ušetřit spoustu drahocenného času. Součástí cvičných souborů čtvrté kapitoly jsou k dispozici dvě takové šablony. Prohlédněte si je.



Obr. 11 Ukázková šablona napsaná v XHTML 1.0



Obr. 12 Ukázková šablona napsaná v HTML5

Abyste ale byli schopni šablonu dále upravovat, a aby splňovala konkrétní účely, měli byste vědět, jak se vlastně taková šablona tvoří a jak funguje. Není to jen samotný layout, ale spousta dalších prvků, které se zobrazují a chovají podle přesného zadání. Proč mluvíme o šablonách, když jsou to v podstatě hotové www stránky? Ano, můžeme tyto šablony označit za statické webové stránky a začít je okamžitě využívat. Šablony se jim říká ze dvou důvodů:

- 1 Lze je dále upravovat (přidávat či měnit obsah atp.),
- 2 Většinou se na ně dále nabalují dynamické skriptovací prvky a vznikají tak dynamické weby.



evropský
sociální
fond v ČR



MSMT
MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Tvorba vlastní šablony

Součástí čtvrté kapitoly je příklad, který si budeme dále popisovat (složka **sablona**). Po otevření přiložené složky máme před sebou velmi jednoduchou, ale vlastně kompletní šablonu menšího webu napsanou v XHTML 1.0.

Složka **sablona** obsahuje soubory XHTML

- index.html,
- info.html,
- kontakt.html,

dále, soubor kaskádových stylů CSS

- style.css,

a podsložky

- soubory,
- a obrázky.

Jedno velmi důležité upozornění. Jistě neuniklo vaší pozornosti, že jsme www stránku už několikrát nazvali **index**. Neučinili jsme tak jen náhodou, stejně jako v tomto případě.

Každý web má nějakou úvodní stránku. Když ve svém internetovém prohlížeči zadáte třeba adresu <http://www.kteiv.upol.cz>, tak server na kterém jsou tyto stránky umístěny nejprve hledá tuto úvodní stránku, kterou pozná podle názvu **index**. Pakliže ji najde, obratem její obsah zašle čekajícímu žadateli. V případě, že tuto stránku nenajde, zahlásí chybu! A to nehledě na skutečnost, že se zde nachází spousta jiných webových stránek. **Proto i vaše úvodní stránka musí být pojmenována index.** Pochopitelně s danou příponou.

Kódování

Každý HTML dokument má text překládaný v určitém **Kódování**. O jaké kódování se jedná, se internetovému prohlížeči sděluje v hlavičce dokumentu, pomocí následujícího řádku (metatagu):

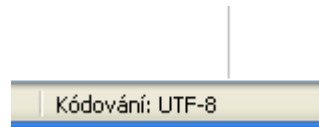
```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
```

V každé stránce musí být kódování nastaveno a pokud možno pro celý web jednotně. V uvedeném příkladu jsme použili kódování **utf-8**, ale existují i jiné.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Není to ale jediná podmínka. Metatag je informace pro prohlížeč a tato informace se musí shodovat s „fyzickým“ kódováním vašich dokumentů. Toto kódování za vás provede např. program **PsPad**. Stačí si u každého dokumentu prohlédnout v *zápatí programu* jaké je nastaveno kódování. Viz další obrázek. Pokud se neshoduje s předepsaným kódováním v metatagu, stačí zvolit záložku **Formát**, a aktivovat příslušné kódování. Poté již jen tento krok potvrdit uložením dokumentu (Ctrl + S).



Obr. 13 Kódování dokumentu v PSPadu

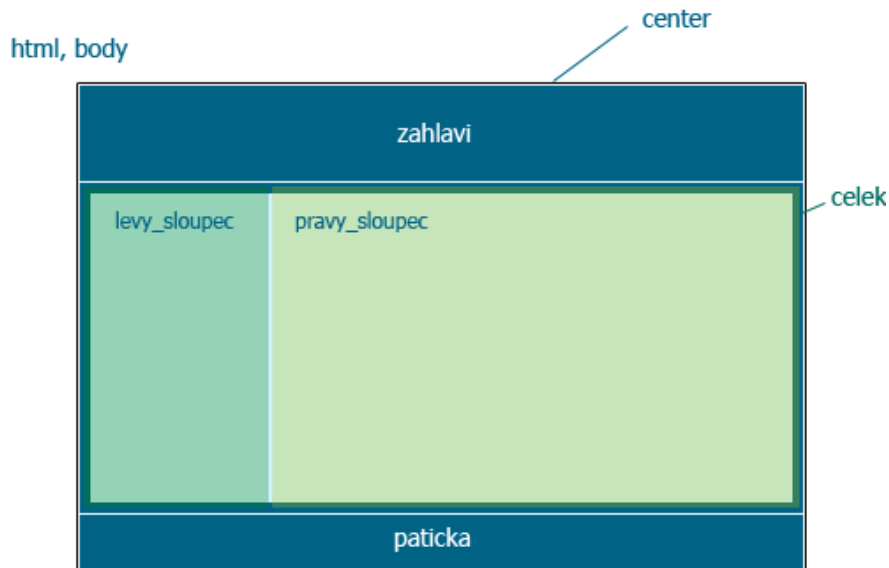
Layout naší šablony je složen z:

- *záhlaví*,
- obsahového divu *celek*,
- *levého sloupce*,
- *pravého sloupce*
- *a patičky*.

Div celek má jeden prostý úkol. Drží levý a pravý sloupec pohromadě. V praxi se toho využívá zejména v případech, kdy chcete pro dva nebo více sloupců jednotné pozadí.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický



Obr. 14 Schéma Layoutu

Abychom se v souboru *style.css* lépe orientovali, divy **layoutu** jsem zapisoval následujícím způsobem s vlastnostmi pod sebe:

```
div#levy_sloupec {
    width:250px;
    margin:0;
    float:left;
}
```

Dílčí selektory, např. pro odstavce, obrázky nebo nadpisy náležící danému bloku layoutu jsem zapisoval s vlastnostmi do řádku:

```
#levy_sloupec img {margin:1px 0 0px 0; border:0; width:250px;
    height:105px;}
```

Layout

Nastavení selektoru **html, body** již zde nebudu popisovat, měli byste jej zvládnout z kapitoly 3. Zajímavější je situace kolem záhlaví naší šablony. Na pozadí je totiž umístěn obrázek:



Obr. 15 Pozadí záhlaví layoutu

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Stačí si takový obrázek vytvořit v některém grafickém programu, uložit do příslušné složky našeho webu a potom se na něj pomocí vlastnosti **background** v CSS odkázat.

```
background: url(obrazky/zahlavi.png) top left no-repeat;
```

Hodnotou **url** určíte cestu k obrázku (resp. pozadí). Dále nastavíme vertikální zarovnání (**top** nebo **bottom**), horizontální zarovnání (**left** nebo **right**) a určíte, jak se má pozadí na stránce opakovat.

Vzhledem k tomu, že pozadí vyplní celý div záhlaví (tj. obrázek na pozadí má stejnou šířku a výšku jako samotný div), není potřeba opakování. Takže můžeme nastavit **no-repeat**. (**jinak může být opakování prováděno podél osy X repeat-x nebo Y repeat-y**). Další nastavení záhlaví se již týká okrajů, obtékání, nastavení šířky a výšky.

Přichází na řadu **div celek**. O jeho významu už víte. Důležitá je jeho šířka. Nastavili jsme maximální šířku – tedy **900px** (obrazových bodů).

levy_sloupec a **pravy_sloupec** jsou vnořeny do divu **celek**. Součet jejich šířky spolu s okraji **nesmí (neměl by) přesáhnout** šířku divu **celek**.

Dalším prvkem našeho layoutu je **levý sloupec**. Zvolili jsme pro něj nějakou šířku a nastavili okraje + obtékání¹². Do tohoto levého sloupce již chceme vložit nějaký obsah. Pokud jste si již prohlédli šablonu v internetovém prohlížeči, víte, že je v něm umístěn obrázek a menu našich stránek. Pro všechny tyto elementy jsme nastavili příslušné vlastnosti.

Před každý dílčí selektor u levého sloupce píšeme **#levy_sloupec** selektor {}

Např.

```
#levy_sloupec img {}  
#levy_sloupec p {}  
...
```

Činíme tak z důvodu, že dané vlastnosti pro odstavec **p** nebo obrázek **img** se budou týkat pouze těch HTML elementů (tagů), vnořených do levého sloupce. Např. pro odstavec **p** v pravém sloupci už budou platit jiné vlastnosti, deklarované:

```
#pravy_sloupec p {}
```

Můžeme tak oddělit vlastnosti stejných elementů (tagů) v různých blocích layoutu.

Pokud bychom chtěli pro celou webovou stránku jednotné nastavení elementu, třeba odstavce, zapíše se na začátek v souboru CSS rovnou do tvaru

¹² vnější okraje **margin**, obtékání **float**

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

p {}

nebo

body p {}

Výše zmíněným způsobem jsme tak postupně nastavili v levém sloupci formátování obrázku, odstavce a odkazu (viz. soubor *style.css*).



Obr. 16 Šablona XHTML

Dědičnost stylů

V levém sloupci našeho layoutu jsme potřebovali vzhledem k tvorbě menu využít změnu formátování odstavce. Abychom graficky odlišili odstavec s textem „Nabídka“ od odstavců v menu stránek. Menu stránek se sice obvykle tvoří pomocí seznamu (TAG ul), ale my jsme pro ukázkou zvolili další použití formátovaného odstavce. Proto jsme si vytvořili odstavec s **třídou menu**:

```
#levy_sloupec p.menu {}
```

Na stránce se na něj pak odkazujeme `<p class="menu"></p>`. Ve stylech CSS nám pomáhá jejich **dědičnost**. Tzn., že do bloku deklarací odstavce s třídou menu musíme uvést pouze ty vlastnosti, které chceme z rodičovského elementu (je-li nastaven) změnit. Česky řečeno, uvedeme pouze ty vlastnosti, které mají mít odlišnou hodnotu:

```
background:#c5e6f4;
```

```
color:#000;
```

```
font-weight:normal;
```

Prohlédněte si komentář v souboru style.css.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Všimněte si rozdílu v internetovém prohlížeči. Jiné pozadí, barva písma a tloušťka písma. Ostatní vlastnosti, které jsem neuvedl, odstavec s třídou menu zdědí po nadřazeném (rodičovském) odstavci levého sloupce bez třídy. Tj:

```
#levy_sloupec p {
    margin:1px 0 1px 0;
    padding:5px 0 5px 5px;
    border-bottom:1px solid #7a7a7a;
    font-size:12px;
}
```

Pokud se ještě v dědičnosti stylů CSS neorientujete, zkuste si to představit následovně. Syn dědí po otci v ideálním případě všechny vnější fyzické vlastnosti.

```
#otec {
    Výška: 180;
    Hmotnost: 80;
    Barva vlasů: Černá;
}
#otec .syn {
    Barva vlasů: Zelená;
}
```

V našem případě je tedy syn vysoký 180, váží 80, ale vlasy si zřejmě přebarvil 😊.

Odkazy

Hypertextové odkazy se formátují ve třech úrovních. Jak má vypadat **aktivní odkaz a:link**, **navštívený odkaz a:visited**, a **myši detekovaný a:hover**.

Pokračování

U pravého sloupce a u patičky jsme postupovali úplně stejně. Deklarovali jsme opět nastavení potřebných elementů pro pravý sloupec a pak pro patičku. Pochopitelně můžeme další selektory libovolně přidávat.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Kompatibilita zobrazování a funkčnosti

Nejpoužívanější int. prohlížeče dnes jsou **IE 9** (10 a 11) a **Mozilla Firefox**, silně dotahuje **OPERA**, **CHROME** a **Safari**. Je potřeba minimálně pro první tři uvedené prohlížeče zajistit kompatibilitu našich stránek.

Testujte proto zobrazení vašich šablon v různých prohlížečích a hledejte případné nedostatky.

Úkol

Prostudujte si, jaký je rozdíl mezi vlastnostmi CSS **margin** a **padding**.

Jak Psát Web: <http://www.jakpsatweb.cz/okraje.html>

Shrnutí

- Kaskádové styly CSS slouží k formátování vzhledu www stránky.
- Nejčastěji se zapisují do externího souboru, na který se odkazuje z www stránky v hlavičce pomocí nepárového tagu **<link />**.
- Můžeme využít jeden soubor CSS pro více dokumentů.
- **CSS** umožňuje tvorbu layoutu.
- Vlastnosti CSS se mohou nastavit pro kterýkoli element stránky přímo, nebo pomocí identifikátorů a tříd.
- **Identifikátor** pro daný element se může na stránce použít pouze jednou (vhodný pro layout), **třída** i vícekrát.
- Barvy se jako hodnota vlastností zapisují buď anglickými názvy (pouze 16 základních) nebo pomocí **#** a **hexadecimálního tvaru** dané barvy. Např. barva bílá #ffffff. Barva černá #000000. *Lze využít i zápisu pro 3 základní složky RGB.*
- Kaskádové styly CSS umožňují vkládání obrázků na pozadí jednotlivých elementů.
- Kaskádové styly podporují dědičnost.

Pojmy k zapamatování



Layout, Šablony, Dědičnost CSS

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Samostatně

V šabloně (složka **sablona**), se kterou jsme pracovali v rámci 4. kapitoly, jsou pod odkazy **informace** a **kontakt** jednoduché úkoly.

V sekci **informace** je vložena tabulka, která ale hyzdí svým vzhledem vše ostatní. V souboru *style.css* již máte přichystány **selektory**:

```
#pravy_sloupec table {}
```

```
#pravy_sloupec tr {}
```

```
#pravy_sloupec th {}
```

```
#pravy_sloupec td {}
```

Do nich je potřeba doplnit vlastnosti a tabulku formátovat. Je na vás, jak tabulku upravíte. Do HTML dokumentu vlastně není potřeba zasahovat. Stačí jen doplnit vlastnosti v souboru CSS. Pokud budete chtít vytvářet nějaké třídy, např. pro jednotlivé buňky **<td>** budete již muset zasáhnout i do HTML dokumentu. Úkol je dále popsán přímo v dané šabloně, pod odkazem **informace**.

V sekci **kontakt** máte další úkol, který je tam podrobně specifikován.

Průvodce studiem



Se čtvrtou kapitolou končí naše práce na tvorbě statických HTML dokumentů a šablon a dále se již podíváme na ty součásti, které dávají webovým stránkám možnost komunikovat s uživateli, možnosti stát se dynamickými.

Pakliže jste v předchozí problematice nenašli zalíbení, máte k dispozici free šablony. Není hanbou je využívat, naopak, mnohdy je jejich použití přímo optimální.

Důležité



Nastudujte si přílohu této opory – **BOOTSTRAP**.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

5 Webové formuláře

Cíle

Po prostudování této kapitoly budete:

- umět začlenit formulář do obsahu www stránek,
- umět využít externí podpory pro implementaci formulářových prvků a zpracování zasílaných údajů.

Doba potřebná ke studiu

Na prostudování kapitoly Vám bude stačit **60 minut**.

Formuláře jsou dnes součástí většiny webů. Nejčastěji se setkáme s emailovými formuláři, registračními formuláři nebo formuláři umožňujícími vložit náš příspěvek do diskuze. Formuláře jsou rozhraním v rámci komunikace uživatele s webovou aplikací.

Základy formulářů

Obsahem formulářů jsou různé funkční prvky, které umožňují uživateli zadávat nejrůznější informace.

Formuláře zahrnují textová a zaškrťovací pole, rolovací seznamy, tlačítka, přepínače atd. Vytvořit takovýto formulář není pro člověka znalého základů HTML a CSS žádný problém. Problém nastává až v momentě, kdy se data získaná formulářem mají někam odeslat nebo uložit, případně nějakým způsobem zpracovat. **A na tuto činnost je již HTML krátké.** Je potřeba znalosti programovacích jazyků, pomocí nichž se definují další operace, které se získanými údaji z formulářů manipulují. Např. odesílají data na email adresáta, ukládají data do úložiště (soubory, databáze) nebo pomocí těchto dat filtrují výsledky zobrazení produktů eshopů.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Please complete the form below. Mandatory fields marked *

Delivery Details	
Name *	<input type="text"/>
Address *	<input type="text"/>
Town/City	<input type="text"/>
County *	<input type="text"/>
Postcode *	<input type="text"/>
Is this address also your invoice address? *	
	<input type="radio"/> Yes
	<input type="radio"/> No

Obr. 17 Ukázka webového formuláře

Zabývat se představením obsáhlé problematiky programování webových aplikací už ale značně přesahuje rámec této studijní opory a proto se na něj zaměříme v dalších kapitolách jen ve vybraných ukázkových řešeních. Zájemci, kteří by se rádi s touto problematikou seznámili už nyní detailněji, mohou se odrazit od obsahu následujících odkazů:

Seriál věnující se obecně problematice formulářů:

Interval – Seriál o formulářích <http://interval.cz/clanky/xhtml-formulare/>

Základy zpracování dat odesílaných pomocí formulářů v jazyce PHP

Jak Psát Web – Možnosti PHP <http://www.jakpsatweb.cz/php/moznosti-php.html>

Formuláře v praxi

Formulář je ohraničen v obsahu HTML dokumentu párovým tagem `<form>`. Ukažme si jednoduchý příklad. Následuje HTML kód formuláře a na obr. výstup v prohlížeči:

```
<form method="post" action="skript-zpracovani.html"
enctype="text/plain">
  <input type="text" name="jmeno" />
  <input type="submit" name="odeslat" value="Odeslat jméno" />
</form>
```



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Obr. 18 Ukázka formuláře z předchozího zdrojového kódu

Tag **<form>** obsahuje 3 důležité atributy:

METHOD – Určuje, jakou metodou mají být dále předána data z formuláře.

1. Metoda **GET** – předává data pomocí URL. Obsah formulářových polí se předá jakou součástí adresy URL, která může vypadat takto:

`http://www.example.cz/skript-zpracovani.html?jmeno=honza&odeslat=Odeslat%jméno`

Obsah formulářových polí (hodnota zadaná uživatelem, nebo přednastavená hodnota atributu **value**) se načte do tzv. superglobální proměnné v poli **GET**, odkud je dostupná pro další zpracování.

Tato metoda není ideální pro všechny případy použití, protože se data odeslaná formulářem přenášejí viditelně v URL a podléhají tak určitému bezpečnostnímu riziku. Taktéž množství dat přenesených touto metodou je značně omezené.

2. Metoda **POST** - je vhodná pro obsáhlejší data. Odesílaná data jsou skryta a uživatel je nemůže ve výsledku modifikovat jako v případě metody GET (POST bezpečnější oproti metodě GET).

ACTION - Kam mají být data odeslána (cesta k souboru, který soubor je bude zpracovávat).

ENCTYPE - zakódování obsahu (text/plain | multipart/form-data | application/x-www-form-urlencoded).

Vše co je umístěno mezi párovými značkami tagu **<form>** je obsahem formuláře. Již bylo řečeno, že obsahem formuláře mohou být různé prvky.

Atributy tagu FORM		
Atribut	Význam	Hodnoty
action	skript, který bude zpracovávat data	URL
method	způsob předávání dat	get post
enctype	způsob zakódování dat	application/x-www-form-urlencoded multipart/form-data libovolná mime deklarace

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Prvky formuláře

Obsáhlý a přehledný popis prvků formuláře je k dispozici online na následujícím linku. Já jej zde neuvádím, a proto vás na něj pouze nasměřuju. **Ovšem rozhodně jej důkladně prostudujte:**

Jak Psát Web – Formuláře – <http://www.jakpsatweb.cz/html/formulare.html>

Prakticky si vyzkoušet práci s formuláře budeme moci v rámci 6.kapitoly, kde je rovnou propojíme s klientskou skriptovací technologií.

Služby třetích stran

Nabízí se otázka, proč se vůbec věnovat formulářům, když bez znalosti programování jsou vlastně k ničemu? Přeci jen jedno opodstatnění to má. Můžete totiž využít fungující formuláře i bez potřebných znalostí programovacích jazyků. Stačí jen vědět, jak na to. Využít například **služeb třetích stran** ale znamená, mít alespoň základní přehled o tom, co daný formulář vlastně umí.

Řekněme, že byste rádi na svém webu měli emailový formulář, aby vám návštěvníci vašich stránek mohli zasílat vzkazy, dotazy atp.

1. Na internetu vyhledáte server, který se tímto zabývá.
2. Zdarma se zaregistrujete.
3. Vyberete si službu, o kterou byste měli zájem.
4. Provedete příslušné nastavení a server vám poskytne jednoduchý HTML kód, který jen prostě vložíte na zvolené místo vašich www stránek (CTRL-C, CTRL-V), a je hotovo.

Ukázka služby



Váš e-mail

Vzkaz

Miniaplikace Blueboard.cz

Obr. 19 Emailový formulář ze serveru Blueboard

Na vašich webových stránkách bude umístěna (zobrazena) miniaplikace, jejíž činnost je spravována na onom vzdáleném serveru. A vy máte po starostech.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Jedním z takových serverů je například:

Miniaplikace BlueBoard: <http://miniaplikace.blueboard.cz/>

Shrnutí

- Webové formuláře zajišťují interaktivní komunikační vazbu mezi různými odesílateli a příjemci.
- Webové formuláře jsou ohraničeny párovým tagem <form> a mohou obsahovat různé vstupní ovládací prvky.
- Metody předání odesílaných dat webovým formulářem jsou GET a POST.
- Webové formuláře lze využívat za pomoci třetích stran v rámci služeb různých serverů.

Pojmy k zapamatování



Fomulář, Metody zpracování formuláře, GET, POST, Formulářové prvky, Služby třetích stran

Samostatně

Navštivte server **BLUEBOARD** a prohlédněte si služby, které tento nebo podobný server pro uživatele nabízí.

Průvodce studiem



Pátá kapitola nás uvedla do problematiky tvorby dynamických a interaktivních www stránek, které využívají formulářů. Dále se podíváme nejprve na klientský skriptovací jazyk JavaScript a jeho možnosti.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

6 JavaScript

Cíle

Po prostudování této kapitoly budete:

- znát zásady použití jazyka JavaScript,
- umět přikládat skripty jazyka JavaScript do HTML dokumentů,
- umět vytvořit jednoduchý skript pro validaci odesílaného formuláře,
- využívat JS frameworků a knihoven jako např. LightBox nebo jQuery,
- znát základy Document Object Model (DOM).

Doba potřebná ke studiu

Na prostudování kapitoly Vám bude stačit **300 minut**.

Úvod

JavaScript je klientský **skriptovací jazyk**, který se používá na webových stránkách.

Klientský = **program (kód napsaný v JavaScriptu) se odesílá přímo na klienta** (do prohlížeče) a teprve tam je vykonáván. (Protikladem klientských skriptů jsou skripty serverové, které jsou vykonávány na serveru a na klienta jdou už jen výsledky.)





evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

JavaScript je jazyk:

- ✓ **interpretovaný**¹³ (zpracovává jej samotný int. prohlížeč)
- ✓ **objektový** - využívá interních objektů prohlížeče a zabudovaných objektů, lze tvořit objekty vlastní
- ✓ **závislý na prohlížeči** - funguje ale ve většině prohlížečů
- ✓ **case sensitivní** - záleží na velikosti písem v zápisu zdrojového kódu
- ✓ syntaxí podobný jazykům C, Java a podobným

Prosím mějte na paměti, že JavaScript ≠ JAVA.

Omezení:

- JavaScript funguje pouze v prohlížeči.
- Uživatel může JavaScript zakázat.
- Existují různé odlišné verze jazyka i prohlížečů, což vede k častým chybám.
- Neumí přistupovat k souborům (kromě cookies) ani k žádným systémovým objektům.
- Neumí žádná data uložit (kromě cookies).

Použití

JavaScript je všestranným jazykem a díky **HTML 5** a **DOM** může na www stránce plnit spoustu úloh. Zmíníme ty základní:

- ✓ **Reagování na události** (např. kliknutí myši, přejetí myši přes objekt, stisknutí klávesy, odeslání formuláře, zavření či otevření nového okna a další).
- ✓ **Manipulace s obsahem HTML elementů** (např. vkládání nového textu do elementu odstavce).
- ✓ **Manipulace se styly HTML elementů** (např. změny barvy varovného hlášení).
- ✓ **Validace dat od uživatele** (kontrola údajů zadaných do formuláře na straně klienta).

¹³ Programovací jazyky členění: Více si k této problematice vyhledejte sami!



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Vložení JavaScriptu do stránky

Je to v podstatě velice jednoduché. Zatím si stále vystačíme se znalostmi základů HTML.

- Skripty se zapisují do HTML mezi tagy `<script>` a `</script>`. Všechno, co je mezi těmito tagy, je program psaný v jazyce JavaScript.
- V příkladě níže je použit příkaz **document.write()**. Ten způsobuje zápis do proudu dokumentu. Zapsaný text se ihned zobrazí v prohlížeči.
- Pokud se zapisuje normální text, musí se obalit uvozovkami (na rozdíl od **proměnné**). Mezi uvozovkami se **nesmí zalomit řádek**.
- Každý příkaz JavaScriptu se ukončuje středníkem;

```
<body>
  <p>Toto je normální text stránky.<br /></p>

  <script type="text/javascript">

    document.write("A toto napsal JavaScript");

  </script>
</body>
```

Možnosti zápisu JavaScriptu v HTML dokumentu

Existují **3 způsoby** zápisu (vložení) JavaScriptu do HTML dokumentu:

1. Pomocí párového tagu **<script>** do proudu dokumentu. (viz předchozí ukázka)
2. Odkazem na **externí soubor**.

```
<script type="text/javascript"
src="lightbox/js/lightbox.js"></script>
```

Odkazy na Externí soubory JavaScriptu (mají příponu **.js**) se umísťují buď do hlavičky HTML dokumentu, stejně jako např. styly CSS, nebo na konec těla HTML dokumentu, těsně před koncovou značkou tagu `<body>`. V praxi se více upřednostňuje druhý způsob, tj. na konec těla dokumentu. Kód JavaScriptu tak bude bezpečně vykonán, až jakmile se uživateli načte (vykreslí) samotná stránka. Stránka navíc uživateli nabídne často **rychlejší odezvu**. V probíraných příkladech (cvičných



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

souborech) sice umísťujeme odkazy na soubory JavaScriptu do hlavičky dokumentu, ale děláme tak spíše z důvodu přehlednosti v rámci naší výuky. V praxi doporučujeme onen druhý způsob.

3. In-line zápis - "v řádku". **Nevyužívá tag <script>**, ale zapisuje se jako atribut jiného tagu. Většinou reaguje na nějakou *událost*.

Příklad:

```
<a href="http://www.upol.cz" onmouseover = "alert('Na univerzitě je myš!!!')">UPOL</a>
```

- Příklad ukazuje klasický odkaz s textem UPOL. Důležitý je **atribut (událost) onmouseover**. Při přejetí myši se vykoná příkaz **alert**.¹⁴
- Všimněte si, že v in-line zápisu je nutno používat apostrofy místo uvozovek (protože uvozovky by zakončily zápis skriptu).
- Onmouseover je jednou z událostí, kterých existuje asi tucet. Bez existence událostí by in-line zápis skriptu nedával smysl.

Inline zápis se již nedoporučuje používat. Nejde o to, že by nebyl podporován, ale jde o nežádoucí míchání obsahu HTML dokumentu a aplikační logiky. Význam je podobný jako v případě CSS. Kaskádové styly se rovněž nedoporučují zapisovat do stránky, ale do externího souboru. Nedochozí tak k míchání obsahu, od jeho formátování.

Příklady

Cvičné soubory k 6. kapitole tvoří složka **formulare**. Jedná se o cvičné www stránky, které obsahují prvky probírané v této kapitole. V dalším textu vás budu odkazovat na jednotlivé části těchto stránek. Vyzkoušejte vždy funkčnost a rovněž prozkoumejte i zdrojový kód, který obsahuje i některé důležité a doplňující komentáře.

Prvky a syntaxe jazyka

V souboru JavaScriptu (.js) se zapisuje zdrojový kód přímo, bez uvození do tagu <script>. Stejně tak si budeme představovat následující prvky jazyka.

Základní syntaxe

Příkazy se oddělují středníkem nebo koncem (zalomením) řádku.

první příkaz;

druhý příkaz;

¹⁴ Na úvodní stránce webu **formulare** najdete odkazy na přehled možných událostí, a lze si je i prakticky vyzkoušet.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Záleží na velikosti písmen v proměnných i v názvech objektů.

Objekt **window.Location** je špatně, protože správný tvar je **window.location**.

Řetězce se uzavírají do uvozovek

Je možné využívat i apostrofů, které jsou i nezbytné při in-line zápisu.

```
<img onmouseover="alert('AHOJ')" />
```

Escapování znaků

Pro escapování znaků se používá zpětné lomítko .

```
<img onmouseover="alert(\"AHOJ\")" />
```

Logické hodnoty

JavaScript zná speciální hodnotu pro pravdu: **true** a pro nepravdu: **false**.

Další pravidla

Objekty a jejich metody a vlastnosti se oddělují tečkami: **objekt.podobjekt.vlastnost**

Programové sekvence se uzavírají do složených závorek **{}** (zejména při větvení a deklaraci funkcí).

Proměnné

V programování představují **proměnné** „úložiště“ informace (vyhrazené místo v paměti).

```
var x = "Obsah proměnné";  
document.write(x);
```

- Názvem proměnné může být libovolné slovo (s výjimkou několika klíčových slov jazyka). Záleží na velikosti písmen.
- V JavaScriptu je vhodné proměnné deklarovat. Na deklarování slouží klíčové slovo **var** následované výpisem deklarovaných proměnných.
- Proměnná použitá v příkladu má název **x**. Do proměnné **x** je načtena hodnota "*Obsah proměnné*", která je dalším příkazem **document.write** zapsána do proudu dokumentu.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Příklad:

```
var y, text;  
y = 13;  
y = y * 2 * 7 * 11;  
text = "Výsledné číslo je" + y;  
document.write(text);
```

Co vypíše příkaz **dokument.write(text)**?

Operátory

1. Operátory přiřazení

=	přiřazení
+=	přičtení, ale také připojení řetězce
*=, -=, /=	přinásobení, odečtení, "přidělení,,
++	přičtení 1
--	odečtení 1

Zápis	Ekvivalentní tvar zápisu
a += 4	a = a + 4
b++	b = b + 1

2. Početní operátory

+	sčítání i spojování řetězců (příklad cyklu for níže)
-	odčítání, unární negace
*	násobení
/	dělení

3. Logické operátory

==	rovnost (dvě rovnítky)
!=	nerovnost

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

<, <=, >=, >	aritmetické srovnání
&&	logické AND (a zároveň)
	logické OR (nebo)
!	logické NOT (negace)
?:	podmínkový výběr (ternární operátor)
,	logické spojení (třeba v zápisu parametrů funkcí)

Logické operátory se používají zejména při větvení programů na stanovení podmínek. Podmínky nabývají hodnot **true** (pravda) a **false** (nepravda).

Větvení

Větvení nebo-li **podmínkové konstrukce** jsou klasickou součástí téměř všech programovacích jazyků. Struktura této konstrukce ve většině programovacích jazyků téměř shodná. I z toho důvodu je kombinace JavaScriptu a PHP velmi oblíbená, protože naučíte-li se základům jednoho z nich, u druhého máte z poloviny vyhráno.

Klauzule if

```
if (podmínka1) {  
    příkazy prováděné při splnění podmínky;  
}  
  
else if (podmínka2) {  
    příkazy prováděné při nesplnění podmínky1 a splnění podmínky2;  
}  
  
else if (podmínkan) {  
    příkazy prováděné při nesplnění podmínek(n-1) a splnění podmínkyn;  
}  
  
else {  
    příkazy prováděné při nesplnění všech n předchozích podmínek;  
}
```

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Příklad:

```
Var znamka;  
znamka = 2;  
if (znamka == 5) {  
    document.write("To není moc hezká známka");  
}  
else {  
    document.write("Hlavně že to není za 5");  
}
```

Cykly

Cyklus je v programování **konstrukce**, která dle stanovených podmínek může opakovat provádění příslušných operací. Jednoduchý příklad může zahrnovat rozesílání spamu. Program rozesílá nechtěné reklamy. Rozeslání se provede tolikrát, kolik program nalezne emailových adres. A šťastní adresáti se mají nač těšit.

Cyklus WHILE

Cyklus s **podmínkou na začátku**. Sekvence vnitřních příkazů se provádí dokola, dokud platí podmínka. Jakmile podmínka neplatí, vnitřek příkazu **while** už se nevykoná a program bude pokračovat pod koncem sekvence while.

```
while (podmínka) {  
    sekvence příkazů  
}
```

Následující příklad zvyšuje proměnnou o pět jednotek, dokud nebude větší nebo rovna třiceti:

```
Var x;  
while (x < 30) {  
    x += 5;  
}
```

Kolik opakování (iterací) cyklus v předchozím příkladu proveden?

Cyklus FOR

Cyklus určený **pro daný počet opakování**.

```
for (počáteční hodnota; podmínka; navýšení) {  
    příkazy;  
}
```

Např. tento program vypíše všechny HTML nadpisy:



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

```
for (i=1; i <= 7; i++) {  
    document.write("<h" + i + ">Nadpis</h" + i + ">");  
    document.write("<br />");  
}
```

Funkce

V JavaScriptu existuje podobně jako v jiných programovacích jazycích možnost definovat (deklarovat) vlastní funkce. Funkce je programový blok, který může (ale nemusí) zpracovávat vstupní n-vstupních parametrů a vrací určitou hodnotu (pomocí klíčového slova **return**). Výhodou funkcí je jejich znovupoužitelnost, což umožňuje eliminovat mnoho duplicitního kódu.

Deklarace funkce

```
function jmenoFunkce (parametr1, parametr2, ...,parametrn)  
{  
    příkaz;  
  
    příkaz;  
  
    return hodnota  
}
```

Volání funkce

```
jmenoFunkce (hodnota, hodnota);
```

Příklad funkce v rámci validace formuláře

```
function kontrolaJmena (vstup) {  
    var zadanaHodnota;  
        zadanaHodnota =  
vstup.elements.namedItem("name").value;  
    if ( zadanaHodnota == "" ) {  
        alert ("Zadejte prosím vaše jméno");  
        return false;  
    }  
    else if ( zadanaHodnota == "Napiš jméno" ) {  
        alert ("Zadejte prosím vaše jméno");  
        return false;  
    }  
    else return true;  
}
```



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Cvičení

Použití vlastních funkcí si můžete prohlédnout a procvičit na stránkách **Kontrolovaný formulář** a **Plně kontrolovaných formulář** webu formulare.

Více informací k JavaScriptu a zejména k jeho objektovému modelu najdete na webu **W3Schools.com**: <http://www.w3schools.com/js/default.asp>

Hlášení

JavaScript používá tři druhy boxů hlášení:

1. Upozornění **alert**.
2. Dialogové okno – vstup pro uživatele **prompt**.
3. Poptvrzovací okno **confirm**.

Ukázky a zdrojový kód si můžete prohlédnout třeba na stránce JS HTML DOM.

Objektové vlastnosti JavaScriptu

Již výše jsme si řekli, že JavaScript je jazyk objektový. V praxi to pro případ JavaScriptu znamená, že téměř všechno může být objektem. Např. řetězec, pole, funkce, data atd. Měli byste vědět, že objekt je tvořen:

- 1) **Vlastnostmi** – proměnné asociované s objektem.
- 2) **Metodami** – funkce, které objekt může vykonávat.

Velmi oblíbený příklad, kterým s oblibou demonstrují na přednáškách princip objektu a jeho vlastností a metod si dnes můžeme prohlédnout i na webu W3Schools.com. Mějme tedy objekt **auto**:

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Objekt



Vlastnosti

car.name = Fiat

car.model = 500

car.weight = 850kg

car.color = white

Metody

car.start()

car.drive()

car.brake()

Vlastnosti objektu auto jsou např. jeho název, označení modelu, hmotnost a barva. Všechny auta (objekty) mají tyto vlastnosti, jenom se mění hodnoty těchto vlastností.

Metody objektu auto jsou nastartování, ježdění a brždění. Všechny auta (objekty) mohou vykonávat tyto akce (metody).

Nyní se můžeme podívat, jakým způsobem lze s objekty pracovat přímo v JavaScriptu.

Vytvoření objektu:

V JavaScriptu se např. objekt řetězce vytváří pomocí klíčového slova **new**. Objekt uložíme do proměnné **txt**.

```
var txt = new String("Ahoj Hercule");
```

Vlastnosti:

Tento objekt má např. vlastnost **length** (v ní je uložena hodnota počtu znaků řetězce). Zjištění hodnoty této vlastnosti je velmi jednoduché.

```
txt.length // obsahuje hodnotu 12
```

Metody:

Objekt řetězce má např. metodu **indexOf()**. Ta vrací pozici prvního výskytu hledaného podřetězce.

```
txt.indexOf("Hercule"); // vrátí 5
```

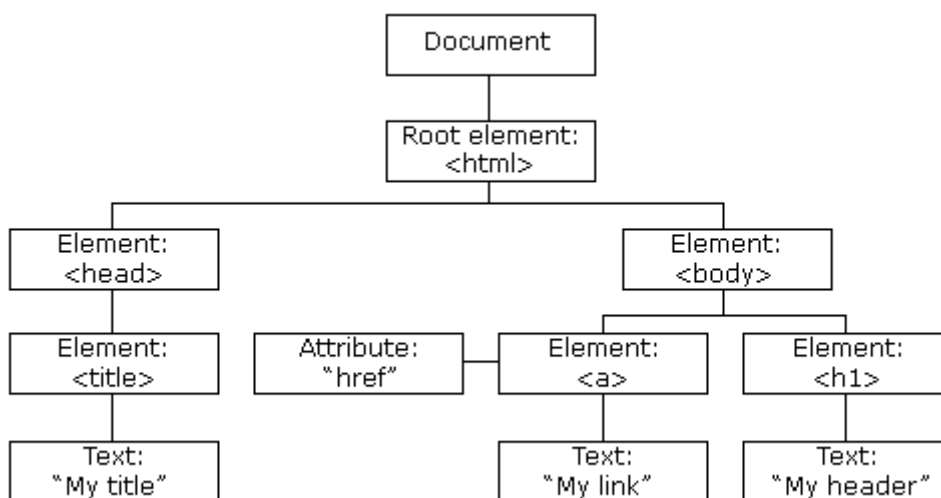
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Document Object Model (DOM)

Díky **DOM** může **JavaScript** přistupovat k libovolným HTML elementům a může je měnit. V momentě, kdy je webová stránka načtena, vytvoří prohlížeč její objektový model. HTML DOM je konstruován jako strom objektů:



Obr. 20 Strom HTML objektů

HTML DOM definuje:

- HTML elementy jako objekty.
- Vlastnosti všech HTML elementů.
- Metody pro přístup k HTML elementům.
- Události pro všechny HTML elementy.

Příklad

Ukážeme si jednoduchou ukázkou použití jedné z metod pro přístup k HTML elementům, a speciální vlastnosti, která umožňuje získat obsah elementu a rovněž jej měnit.

Metoda **getElementById()** je nejběžnější metoda pro přístup k HTML elementům prostřednictvím jejich id. Ano, přesně podle těch id, které jsme se naučili už využívat v rámci použití identifikátorů kaskádových stylů.

Vlastnost **innerHTML** lze použít jak pro získání obsahu zpřístupněného elementu, tak pro možnost jeho změny.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

```
<h2 id="header">...</h2>
<p id="intro">Ahoj Hercule!</p>

<script>

    var txt = document.getElementById("intro").innerHTML;
    document.getElementById("header").innerHTML = txt;

</script>
```

Ačkoli je obsah nadpisu druhé úrovně s identifikátorem header „...“, tak jej jednoduchý skript JS s využitím metod a vlastností DOM nahradí textem z elementu odstavce s identifikátorem intro. Příklad si můžete prohlédnout na stránce JS HTML DOM.

Příklad

S objektovým přístupem a částečně i DOM jsme se již setkali v případě validace formuláře v našem zkušebním webu (**formulare**). Podívejme se blíže na následující řádky kódu (část zdrojového kódu HTML dokumentu *kontrolovany-form.html*):

Volání funkce:

```
<form method="" action="" onsubmit="return kontrolaJmena(this);">
```

V momentě odeslání formuláře (událost **onsubmit**) je ještě před jeho samotným odesláním volána funkce **kontrolaJmena**, které je jako parametr předán objekt zpracovávaného formuláře (klíčové slovo **this**).

Funkce, která provádí validaci, tento parametr označuje jako **vstup** (v případě deklarace funkce, lze parametry pojmenovat libovolně).

```
function kontrolaJmena (vstup) {
    var zadanaHodnota;
        zadanaHodnota =
vstup.elements.namedItem("name").value;
    ...
    ... }
```

Do proměnné **zadanaHodnota** je načtena hodnota vložená uživatelem do políčka formuláře Jméno. Objekt formuláře (**vstup**) obsahuje kolekci elementů (tedy prvků formuláře). V našem případě potřebujeme z této kolekce vybrat element, jehož atribut **name** má hodnotu „name“. K totmu účelu souží metody **namedItem()**.

```
vstup.elements.namedItem("name")
```

Předchozí zápis je vlastně odkazem na objekt textového políčka formuláře. A nyní nám už jen stačí dotázat se tohoto objektu na jeho vlastnost **value**, která v sobě nese skutečný obsah textového políčka formuláře.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

`vstup.elements.namedItem("name").value`

Více k této problematice a spousta dalších příkladů k využití na webu W3Schools.com.¹⁵

Knihovny JavaScriptu

Knihovny funkcí jsou připravené funkce JavaScriptu, které nám výrazně usnadňují jeho použití a přinášejí funkcionalitu navíc. Některé knihovny jsou přímo určeny pro konkrétní účely, jako např. pro vytváření obrazových galerií. Jiné jsou navrženy pro zjednodušení navigace, obsluhu událostí atd.

Lightbox

Lightbox je jedna z mnoha JavaScriptových knihoven, které se používají pro prohlížení obrázků v aktuálním okně prohlížeče. Takové fotogalerie jsou dnes již běžnou součástí a denně se s nimi setkáváme na nejrůznějších webových portálech či stránkách. Jejich použití je velmi jednoduché a závisí pouze na stažení příslušných souborů a umístění do kontextu webových stránek.

Knihovnu Lightboxu si spolu s jednoduchým návodem na její použití můžete stáhnout zde: <http://lokeshdhakar.com/projects/lightbox2/>

Její praktickou aplikaci a popis použití obsahuje rovněž stránka **Lightbox** webu [formulare](#). Z dalších podobných knihoven, které se využívají pro obrázkové galerie, zmíníme např. FancyBox nebo GreyBox.

jQuery

jQuery je knihovna navržena pro jednoduchou navigaci v elementech stránky, manipulaci s nimi, obsluhu událostí a provádění vzdálených asynchronních volání. Oproti knihovně jako je Lightbox tedy nestačí pouze stáhnout příslušný kód a vložit ho do stránky, ale je potřeba rovněž umět vytvořit s pomocí jQuery uživatelský kód, který bude plnit námi požadované akce.



Knihovna umožňuje provádět dvě základní akce:

1. **Selekci** – tj. výběr elementů webové stránky, které připadají zadanému selektoru (jako v CSS).
2. **Modifikaci** – např. přidání elementu, odstranění potomků nebo nastavení stylu CSS nad vybranými elementy.

¹⁵ http://www.w3schools.com/js/js_htmlDOM.asp



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Příklad

Následující kód vybere všechny elementy, které patří do třídy **intro**, a nastaví u uvedených vlastností CSS nové hodnoty:

```
$(".intro").css({"color":"red", ".margin-left":"5px"});
```

Knihovna jQuery je v základu součástí pouze jednoho souboru, který jsme mimo jiné již použili s knihovnou Lightbox, která jej pro své účely využívá. Po stažení tohoto souboru¹⁶ jej vložíme pomocí tagu `<script>` do stránky. Vlastní skripty, které budeme s pomocí této knihovny psát, buď vhodně umístíme do jiného externího souboru nebo přímo do stránky mezi tagy `<script>`. Použití jQuery si můžeme opět prohlédnout v akci na webu formulare.

jQuery nenahrazuje JavaScript, ale zachovává jeho funkcionalitu a rozšiřuje pole jeho možností (jedná se o nadstavbu JS). Ve zdrojovém kódu jQuery tak lze používat i klasický JavaScriptový syntax.

Na závěr této kapitoly si řekněme, co jQuery nabízí v porovnání s klasickými JavaScriptem:

- Kód využívající knihovnu jQuery pro výběr elementů a jejich modifikaci je výrazně kratší.
- Kratší kód je lépe čitelný.
- Psaní jQuery zrychluje vývoj.
- S použitím jQuery je možné vytvářet kód kompatibilní se všemi hlavními prohlížeči.

Shrnutí

- JavaScript je klientský skriptovací jazyk, jehož interpretem je internetový prohlížeč.
- JavaScriptu funguje v internetovém prohlížeči.
- JavaScript je CASE senzitivní – v jeho zápisu záleží na velikosti písmen.
- JavaScript využívá programové konstrukce, jako proměnné, podmínky, cykly, funkce známé z jiných programových jazyků – je podobný i PHP.
- Pomocí JavaScriptu lze validovat formuláře, pracovat se zabudovanými objekty atd.
- Díky DOM může JavaScriptu přistupovat a modifikovat elementy www stránky.
- Pro JavaScript existuje celá řada knihoven, z nichž nejvyužívanější je knihovna jQuery.
- jQuery tvoří nadstavbu JavaScriptu. Obsahuje mnoho nových funkcí, jako jsou animace a pohyb elementů na stránce.
- Hlavní nevýhodou JavaScriptu je, že jej uživatel může v prohlížeči zakázat.

¹⁶ <http://jquery.com/>



evropský
sociální
fond v ČR



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Pojmy k zapamatování



JavaScript, Objektový, Interpretovaný jazyk, Case sensitivní, DOM, knihovny, Lightbox, jQuery

Průvodce studiem



Šestá kapitola obsahuje to nejnmutnější, co bychom pro případ JavaScriptu měli znát. Kdyby to bylo možné, obsahovala by studijní opora na téma JavaScript třeba 300 stran a pořád by to bylo málo. Nicméně v případě JavaScriptu lze opět vyhledat hotová řešení konkrétních úkolů a umět je implementovat do svých stránek.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

7 Úvod do jazyka PHP

Cíle

Po prostudování této kapitoly budete:

- umět připravit běhové prostředí nutné pro použití PHP,
- znát základy skriptovacího jazyka PHP,
- umět vložit PHP kód do obsahu www stránky,
- umět naprogramovat jednoduchou aplikaci ve spolupráci s formulářem.

Doba potřebná ke studiu

Na prostudování kapitoly Vám bude stačit **300 minut**.

Skriptovací webové jazyky můžeme primárně rozdělit na *klientské* a *serverové*. **Klientské** fungují na jednoduchém principu. Spolu s (X)HTML stránkou je internetovému prohlížeči odeslán i kus programového kódu a ten je ve správnou chvíli na cílovém počítači zpracován. Již známe např. JavaScript.

Serverové technologie (např. PHP) jsou založeny na jiném principu. Předně musíme říci, že **internetový prohlížeč neumí PHP kód interpretovat!** Neumí ho přeložit. Proto webový server ve spolupráci s interpretem jazyka PHP nejprve PHP kód interpretuje a až poté stránku odešle klientovi.



Pro testování www stránek mimo veřejný webový server (tj. vyzkoušet u sebe na PC), je potřeba mít nainstalován lokální webový server, spolu s interpretem jazyka PHP. Ten nám PHP stránku přeloží (sestaví do srozumitelného formátu pro internetový prohlížeč) a do prohlížeče už odešle klasické HTML. Více o instalaci lokálního webového serveru v příloze 1. (věnujte proto pozornost nejprve této příloze)

Jazyk PHP

PHP je interpretovaný skriptovací jazyk jehož počátky spadají do roku 1994. Vyvinul se z iniciativy jediného člověka a stal se jedním z nejoblíbenějších programovacích jazyků. Podle zjištěných údajů se PHP v současnosti užívá na desítkách milionů domén po celém světě a v nedávné době vyšla jeho další verze s označením PHP 5.5.8.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Obliba jazyka PHP tkví především v jeho jednoduchosti a podobnosti s programovacím jazykem C. Jeho prostřednictvím lze vytvořit "téměř" vše, co souvisí s tvorbou dynamických www stránek. Mezi nejčastější aplikace vytvořené pomocí jazyka PHP patří zejména:

- internetové obchody, portály i sociální sítě,
- podnikové informační systémy (ať už intranetové nebo internetové),
- diskusní fóra,
- redakční systémy,
- firemní prezentace,
- webových stránek, poštovní či databázoví klienti,
- vyhledávače a katalogy,
- školní weby,
- drobnosti typu počítadla, ankety a mnoho dalších.

Praktické příklady a ukázky

Než se pustíme do zkoušení níže uvedených příkladů, je nutné, abyste si prostudovali přílohu 1 a učinili potřebné kroky na svém studijním počítači. Tj. okopírovali si zdrojové příklady do kořenové složky dokumentů webového serveru, odkud je můžete začít zkoušet.



Všechny příklady obsahují ve zdrojovém kódu komentáře.

Jednoduché příklady

Vložení PHP do stránky HTML

Kód PHP se uvozuje mezi počáteční a koncovou značku následujícího tvaru:

```
<?php
    // zde leží kód PHP
?>
```

Narazí-li webový server na tuto značku, provede příkazy a uživateli zašle příslušně vygenerovanou stránku obsahující pouze čisté HTML.

Stránky, které obsahují kód **php** nemají příponu **.html** ale **.php**

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Výpis textu

Alfou a omegou je výpis textu do obsahu HTML stránky. Používají se dva příkazy: **print** a **echo**.

```
<?php
    print "Print vypíše tento text <br />";
    echo "Echo také vypíše text";
?>
```

Rozdíl mezi oběma spočívá v tom, že pomocí příkazu **print** lze vypsát pouze jeden textový řetězec, **echo** jich může vypsát více.

Ukázku použití zmíněných obou příkazů si můžete blíže prohlédnout v souboru *cviceni-1.php* ve složce **php** k 7. kapitole.

Proměnné

Jazyk PHP nevyžaduje deklarace proměnných a používá tzv. dynamické typování. To znamená, že není potřeba proměnnou deklarovat, ani explicitně určit její datový typ. Interpret PHP toto učiní na základě hodnoty proměnné sám. V jazyce PHP píšeme před názvem proměnné znak **\$**.

```
<?php
    $x = "Nějaký text";
    $y = 4;
    $xy = $x . $y;
    echo $xy;
?>
```

PHP důsledně rozlišuje v názvech proměnných velká a malá písmena (CASE Sensitive). **Za znakem \$ se musí nacházet písmeno nebo podtržítka**, nikdy **ne číslo** ani další znaky. Nejlépe využívat klasických textových názvů – i s využitím velbloudího stylu: [\\$krestniJmeno](#)

Ukázku použití proměnných si můžete blíže prohlédnout v souboru *cviceni-2.php*.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Ověření existence proměnné

Pro testování existence proměnné nebo části pole s daným názvem se používá funkce **isset()**.

```
<?php
$hokus = 42;
if (isset($hokus))
    {print "Proměnná \$hokus existuje.<br />";}
else
    {print "Proměnná \$hokus neexistuje.<br />";}

if (isset($pokus))
    {print "Proměnná \$pokus existuje.<br />";}
else
    {print "Proměnná \$pokus neexistuje.<br />";}
?>
```

=== Výstup na stránce HTML ===

Proměnná \$hokus existuje.
Proměnná \$pokus neexistuje.

Ukázku si můžete prohlédnout v souboru *cviceni-3.php*.

Odstranění proměnné

V případě procedurálního programování platí proměnná až do konce skriptu, případně bloku, v němž byla vytvořena – např. v rámci funkce. Můžeme ji však odstranit explicitně. K tomuto účelu slouží funkce **unset()**.

```
<?php
$hokus =42;
if (isset($hokus))
    {print "Proměnná \$hokus existuje.<br />";}
else
    {print "Proměnná \$hokus neexistuje.<br />";}

unset($hokus) ;
```



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

```
if (isset($hokus))
    {print "Proměnná \$hokus existuje.<br />";}
else
    {print "Proměnná \$hokus neexistuje.<br />";}
?>
```

=== Výstup na stránce HTML ===

Proměnná \$hokus existuje.
Proměnná \$hokus neexistuje.

Ukázku si můžete prohlédnout v souboru *cviceni-4.php*.

Textové řetězce

Pod pojmem řetězec chápeme libovolnou posloupnost znaků. Textový řetězec vkládáme do uvozovek (") nebo do apostrofů ('). Sami asi poznáte, do kterých proměnných byl načten textový řetězec:

```
<?php
$a = 19;
$b = 2.3;
$str1 = "textový řetězec";
$str2 = "Zobrazíme předchozí řetězec <em> " . $str1 . " </em>";
print $str2;
?>
```

=== Výstup na stránce HTML ===

Vypíšeme předchozí řetězec *textový řetězec*.

Ukázku si můžete prohlédnout v souboru *cviceni-5.php*.

Zjištění délky řetězce

Pro práci s řetězci se často vyžadují operace typu „zjistí poslední čtyři znaky“ nebo „vypiš třetí znak od konce“. K jejich provedení potřebujeme znát délku dotyčného řetězce. K tomuto účelu slouží funkce **strlen()**. Jedná se o členskou funkci PHP pro práci s text. řetězci.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

```
<?php
    $retezec = "A vůbec nejhorší jsou trpaslíci";
    $delkaRetezce = strlen($retezec);
    echo "Délka řetězce je $delkaRetezce znaků";
?>
```

=== Výstup na stránce HTML ===

Délka řetězce je 31 znaků.

Ukázku si můžete prohlédnout v souboru *cviceni-6.php*.

Nalezení textu v řetězci

Hledáme-li určitou část znaků v řetězci, můžeme k tomu velmi dobře využít funkci **strpos()**. Další z členských funkcí PHP pro práci s text. řetězci.

```
<?php
    $retezec = "A vůbec nejhorší jsou trpaslíci";
    $pozice = strpos($retezec, "trpaslíci");
    echo "Hledané slovo začíná na pozici $pozice";
?>
```

=== Výstup na stránce HTML ===

Hledané slovo začíná na pozici 25.

Ukázku si můžete prohlédnout v souboru *cviceni-7.php*.

Konstrukce

Podmínky

Jak už víte, podmínkové konstrukce umožňují větvení programu na základě splnění určité podmínky. Je-li tato podmínka splněna, pokračuje tok příkazů první větvi. Není-li, pokračuje běh programu větvi jinou. Syntaxe je shodá jako v případě JavaScriptu.

Při sestavování podmínky se používají porovnávací a logické operátory.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Porovnávací operátor	Význam
==	Rovnost
!=	Nerovnost
<>	Nerovnost
===	Identita (stejný dat. typ)
!==	Nestejnost
>	Větší
<	Menší
>=	Větší nebo rovno
<=	Menší nebo rovno

Kombinace se uplatní v případě že:

Logický operátor	Popis
!	Podmínka není pravdivá
AND	Obě podmínky jsou pravdivé
OR	Pravdivá je alespoň jedna podmínka
&&	Obě podmínky jsou pravdivé
	Pravdivá je alespoň jedna podmínka

```
<?php
$a = 10;
$b = 11;
if ($a == $b)
    {echo "Proměnné se rovnají";}
else
    {echo "Je tady rozdíl mezi proměnnými ".abs($a - $b).";}
?>
```

=== Výstup na stránce HTML ===

Je tady rozdíl mezi proměnnými 1. // funkce **abs()** vrací absolutní hodnotu čísla

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Další ukázky s podmínkovými konstrukcemi si můžete vyzkoušet v souboru *cviceni-8.php*.

Cykly

Mnoho bloků kódu je vykonáváno opakovaně, cyklicky. Někdy je počet opakování určen předem – např. pokud je úkolem spamové aplikace odeslat reklamní email milionu „šťastných“ adresátů, jejichž adresy má spammer v databázi.

Cyklus FOR

Syntaxe

```
for (počáteční příkaz; podmínka; koncový příkaz) {  
    Příkaz, nebo skupina příkazů v bloku  
}
```

Blok příkazů se podobně jako u podmínek musí uzavírat do složených závorek {}.

```
<?php  
    for ($i=1; $i<=5; $i++) {  
        echo "Opakovaný text.<br />";  
    }  
?>
```

=== Výstup na stránce HTML ===

Opakovaný text.
Opakovaný text.
Opakovaný text.
Opakovaný text.
Opakovaný text.

Ukázku si můžete prohlédnout a vyzkoušet v souboru *cviceni-9.php*.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Cyklus WHILE

V tomto cyklu je zajímavá tzv. **postfixová inkrementace řídicí proměnné**. Nejprve hodnotu proměnné vypíšeme a ve stejném příkazu ji zvýšíme.

```
<?php
    $i = 1;
    while ($i <= 5) {
        print $i++."<br />";
    }
?>
```

=== Výstup na stránce HTML ===

```
1
2
3
4
5
```

Ukázku si můžete prohlédnout a vyzkoušet v souboru *cviceni-10.php*.

Cyklus FOREACH

Cyklus se velmi často používá k **procházení hodnot prvků polí**. Při použití klasického cyklu for musíme nejprve zjistit počet prvků pole a poté pomocí podmínky vztahující se k určité proměnné můžeme cyklus řídit. Mnohem výhodnější je užití cyklu **foreach**. Jak vyplývá z jeho názvu, bude vykonán pro každý prvek pole.

```
<?php
    $mesice = array("Leden", "Únor", "Březen");
    foreach ($mesice as $cislo => $mesic)
    {
        echo $cislo. ": ".$mesic. "<br />";
    }
?>
```

=== Výstup na stránce HTML ===

```
0: Leden
1: Únor
2: Březen
```

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Ukázku si můžete prohlédnout a vyzkoušet v souboru *cviceni-11.php*.

Shrnutí

- PHP je serverový interpretovaný skriptovací jazyk.
- K testování PHP je potřeba lokální webový server a interpret jazyka PHP.
- PHP ve verzi 5 a vyšší umožňuje plnohodnotně využít jak metod strukturovaného programování, funkcionálního, tak metod objektového programování.
- Kód PHP uvozujeme do značek `<?php ?>`
- Soubory obsahující kód PHP mají příponu `.php` (např. `index.php`)
- PHP využívá širokou škálu členských funkcí (matematické, řetězcové, práce s datумы)
- PHP má neustále rostoucí dokumentaci – více na <http://cz1.php.net/>
- Pro PHP existuje celá řada výborných frameworků – např. Nette nebo Zend Framework
- PHP umí velmi dobře spolupracovat s různými typy databází, nejčastěji pro projekty s databází MySQL či PostgreSQL.
- PHP je s úspěchem využíván pro velkolepé projekty typu Wikipedia.

Pojmy k zapamatování



PHP, Apache, XAMPP

Samostatně

Kapitola věnována PHP předkládá pouze útržkové základy pro další studium, které byste měli čerpat z přímo specializovaných publikací. Doporučuji především publikace uvedené ve stud. literatuře v závěru.

Nejenom obrázek vydá za tisíc slov. Aby vaše studium bylo efektivnější, součástí cvičných souborů k 7. kapitole jsou ukázky hotových aplikací. Všechny příklady jsou důkladně okomentovány a nové prvky jazyka PHP, se kterými se seznámíte, tak jistě lépe pochopíte.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Příklady

Příklady obsahují následující:

- **Jednoduchá kalkulačka**
 - Zpracování dat odeslaných formulářem
- Upload
 - Ukázka zpracování souboru odeslaného formuláře
- **Anketa**
 - Jednoduchá anketa s ukládáním dat do textového souboru
- **Login**
 - Ukázka přihlašování a ověřování uživatele
- **Web (zjednodušené)**
 - Ukázkový web s využitím funkcí pro vkládání souborů a omezení duplicity dat
- Databáze
 - Ukázka využití spolupráce PHP s databází MySQL
- **Závěrečná práce**
 - Obsahuje i OOP v jazyce PHP

Prostudujte si minimálně **tučně** označené příklady a to nejlépe v pořadí, v němž jsou zde uvedeny. Samozřejmě tyto složky musíte vložit do kořenového adresáře webového serveru a poté k nim přistupujete v internetovém prohlížeči např. ve tvaru: <http://localhost/login/>

Průvodce studiem



V sedmé kapitole jsme si představili skriptovací jazyk PHP, díky kterému můžeme naše statické www stránky oživit i na straně serveru. V příkladech jste se mohli naučit tvořit jednoduché aplikace, které se díky vašemu dalšímu studiu mohou rozrůstat a zdokonalovat.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

8 Optimalizace (SEO)

Cíle

Po prostudování této kapitoly budete:

- znát základní principy SEO,
- umět využít techniky SEO při tvorbě vlastního webu,
- umět hodnotit míru kvality webu vzhledem k technikám SEO.

Doba potřebná ke studiu

Na prostudování kapitoly Vám bude stačit **60 minut**.

SEO

Tento pojem je v komplexu webových nástrojů relativně nový a do značné míry souvisí s webovým marketingem. Každý z nás jistě někdy něco vyhledával na **Seznamu** nebo **Googlu** (dva největší dostupní vyhledávači u nás, např. Centrum využívá služeb Googlu). Ať už se jednalo o konkrétní zboží nebo prostě jen nějaký hledaný výraz. Vyhledávač nám na náš dotaz zobrazí seznam www stránek, které našemu dotazu víceméně odpovídají. Jejich pořadí, které vyhledávač určí a následně nám zobrazí je ale velmi důležité. Může se jednat o stovky či tisíce stránek, ale nám většinou postačí těch prvních pár výsledků a o ty ostatní už se tolik nezajímáme. A proč vlastně? Logicky, protože odpověď na náš dotaz nám většinou poskytnou již první stránky z výsledků vyhledávání.

Existují 2 důvody, které mají zodpovědnost za pořadí výsledků vyhledávání:

1. **Placené reklamy** – Portály poskytující vyhledávače (např. Seznam) za určitý poplatek zvýhodňují pozice jednotlivých webů pro určitá vyhledávání (klíčová slova).
2. **SEO** – neboli kvalita technického a obsahového vybavení webu.

Je pravdou, že placené reklamy ve vyhledávání na daná klíčová slova nepředběhneme. Ale můžeme při kvalitním zpracování předběhnout všechny ostatní 😊. A ruku na srdce, o to přeci v drtivé většině případů dnes jde. Přilákat více a třeba i náhodných návštěvníků na náš web.

Ale technologie SEO není jen o tomto. Její snahou je též zajistit bezbariérovost webu a takové podmínky, které umožní všem návštěvníkům co největší pohodlí při prohlížení našich stránek.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Definice

SEO (Search Engine Optimization, optimalizace pro vyhledávače) je metodika vytváření a upravování webových stránek takovým způsobem, aby jejich forma a obsah byly vhodné pro automatizované zpracování v internetových vyhledávačích.

Protože většina používaných SEO technik není původní a vychází ze zásad publikování přístupného a bezbariérového webu, jsou tyto techniky přínosné nejen pro počítačové stroje (vyhledávací roboty), ale i pro běžné uživatele.

Metody SEO

Součástí SEO je několik metod, které si teď blíže představíme:

Kvalitní a unikátní obsah

Stránka by měla mít kvalitní a unikátní obsah, pravidelně (v ideálním případě denně) aktualizovaný. Toto je pochopitelně závislé na typu webových stránek. Nebudete přeci denně aktualizovat kontaktní údaje. Aktuální cenu hodinek, které na svém webu prodáváte, ale určitě budete zvyšovat 😊.

Používání (X)HTML tagů podle normových předpisů

Tvůrce stránky by měl používat na správných místech doporučený sémantický i syntaktický zápis značek pro daný jazyk. Např. `<h1>` pro nadpis první úrovně, `` pro zdůraznění, `` pro silné zdůraznění. Vyhledávače takovému textu přiřkládají větší význam. Jestliže ale webový tvůrce například definuje nadpis pomocí velikosti písma odstavce `<p>`, či ještě hůře pomocí řádkového elementu ``, nezjistí vyhledávací robot důležitost nadpisu.

Vzpomeňte si, jak jsem vás dříve upozorňoval na nadpis *h1*. **Nadpis první úrovně musí být v HTML uveden na každé stránce a pouze jednou.** Vyhledávače taktéž velmi dobře táhnou k nadpisům druhé úrovně.

Nesmíme použít nepovolené tagy. Rozhodnete-li se např. pro XHTML 1.0 Strict, není možné používat tagy, které jsou zde nepodporované, jako např. v XHTML 1.0 Transitional. Tomuto tématu jsme se ale již věnovali dříve. Chcete-li si jej připomenout, můžete na následujícím linku:

<http://www.jakpsatweb.cz/html/xhtmll.html>

Používání titulku, nadpisů a popisů (title, h1, alt)

Každá stránka webu by měla mít **originální titulek** (tag `title` v hlavičce) a **nadpis h1**. Nepoužívejte klasické titulky a nadpisy jako *úvodní stránka*, *druhá stránka atp.* Využívejte v max. možné míře klíčových slov. Např. *Chovatelé psů*, *Stavební firma XY, atp.* Atribut **alt** je součástí tagu `` pro obrázek a zapisujeme do něj alternativní text, kdyby se obrázek nezobrazil. Taktéž má svou nezastupitelnou roli.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Krátká a neměnná URL adresa

Uvedení krátké URL adresy podpoří zájem ostatních uživatelů o odkazování na stránku. Příliš dlouhá a lidem nesrozumitelná URL adresa naopak od odkazování spíše odrazuje. Přítomnost klíčového slova v URL adrese může u některých vyhledávacích strojů zvýšit umístění stránky ve výsledcích hledání těchto slov. Místo mezer používejte pomlčku. Např. stránka bude pojmenována *chovatele-psu.html*, titulek bude *Chovatelé psů* a nadpis h1 *Chovatelé psů vítějte*.

Názvy stránek (myšleno názvy souborů *chovatele-psu.html*) byste měli měnit jen v krajních případech. Vyhledávací robot totiž projde vaše stránky pouze jednou za čas a uloží si je do své databáze (tzv. zaindexuje). **Výsledky vyhledávání zadané uživateli jsou pak pouze výpisem robotovy databáze, nikoli časově reálným výsledkem vyhledávání.** Proto při jakékoli změně názvu vašich stránek může robot uživatelům po nějakou dobu nabízet špatný odkaz na vaše stránky a chvíli (někdy i měsíc) trvá, než robot vaši novou stránku navštíví a svou databázi aktualizuje.

Budování zpětných odkazů

Pokud je na stránku odkazováno z jiných webů, zvyšuje se její hodnocení (v případě Google je to PageRank, u Seznamu je to S-Rank). Toho bylo možno dříve zneužívat pro tzv. google bomby, kdy se pomocí velkého množství odkazů s určitými klíčovými slovy dostala na první místo ve výsledcích hledání zcela nesouvisející stránka. Příkladem Google bomby byla klíčová slova *ostudné selhání* (v angličtině), která na prvních místech zobrazila odkaz na web amerického prezidenta George Bushe. Podle posledních pozorování si nyní Google kontroluje, jestli je na stránkách související téma. Jestliže bude například ze stránky o chovu velryb odkazováno na web o pěstování muškátů, nebude mít odkaz takovou váhu. Kvalitní obsah zvyšuje šanci, že bude na web odkazováno z jiných zdrojů.

Používání description, keywords

Vhodným používáním klíčových slov v **metatagu description** získá stránka v hodnocení robota další kladné body. Pokud se klíčová slova dále v obsahu stránky neopakují nebo je jich moc, ztrácí na prioritě. **Metatag keywords** používá k zjišťování relevance z hlavních vyhledávačů pouze Yahoo.

Metatag description by měl obsahovat krátké a výstižné tvrzení.

Metatag keywords 3 až 4 klíčová slova, které se na stránce ale opravdu vyskytují. Jsou součástí textu, nadpisů atp.

Příklad (hlavička stránky *chovatele-psu.html*):

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="cs" lang="cs">
  <head>
    <title>Chovatelé psů</title>
    <meta name="Description" content="Web určený pro chovatele psů" />
    <meta name="KeyWords" content="chovatelé psů, pes, kynologie" />
  </head>
```


INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Neetické metody

Kromě tzv. etických metod (výše uvedené) se některé webové stránky (jejich neetičtí tvůrci) snaží vylepšit svou pozici ve vyhledávacích metodami podvodnými. **Proti takovým technikám se vyhledávače obvykle snaží bránit.** Stránkám, které takové techniky používají, hrozí při odhalení výrazný propad ve výsledcích, avšak přinejmenším krátkodobě takové techniky mohou být úspěšné.

Mezi typické neetické metody patří například:

Spam

Stránka, na kterou vede velké množství zpětných odkazů, je obecně považována za lepší. Někteří tvůrci stránek se pokouší zvýšit počet takových odkazů tím, že odkazy na své stránky umísťují do internetových diskusí, do komentářů k příspěvkům v blozích apod., aniž by tyto odkazy byly na těchto místech vítány či měly společné téma.

Oklamání robotů

Tvůrce stránky se může pokusit o to, aby zobrazoval vyhledávacím robotům jiný obsah, než ostatním návštěvníkům a tím se dostal na přední pozice i se stránkami, se kterými by bez tohoto podvodu neuspěl.

Skrytý text

Vyhledávače berou v potaz při řazení výsledku i frekvenci klíčových slov/frází. Tvůrce stránky si tuto frekvenci může uměle zvýšit tím, že na stránku umístí text, který obsahuje několikrát různá klíčová slova. Ten následně skryje pomocí CSS, nebo ho nechá zobrazit velice malým písmem, aby normálního návštěvníka stránky nerušil.

Součásti SEO

Ačkoli je někdy uváděno trochu stranou, tak součástí SEO je též **validita zdrojového kódu webových stránek**. Validita neboli platnost zdrojového kódu je vlastnost, která je jednou z podmínek přístupného a bezbariérového webu.

Validitu lze zjišťovat vizuální kontrolou tvůrcem nebo efektivněji pomocí **online automatických validátorů**, které dle verze značkovacího jazyka (HTML, XHTML Transitional, XHTML Strict, atd.) kontrolují zdrojový kód a poté provedou vyhodnocení, na jehož základě pak webdesigner provede úpravu anebo se jen poplácá po zádech.

Validátor prostě hledá naše chyby.

Nejprve si prohlédne hlavičku našeho webu a zkoumá deklaraci. Poté kontroluje danou stránku (nikoli celý web) a hledá chyby. Nejčastěji zapomenuté ukončení tagu, nepovolený výskyt daného tagu ve zvolené verzi jazyka, chyby v attributech atp.

Výborný online validátor přímo od organizace **W3C** zajišťující standardy webu můžete vyzkoušet na následujícím linku:

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

<http://validator.w3.org/>

Zadejte libovolnou URL adresu webu, který chcete validovat a ihned si můžete prohlédnout výsledky.

Komplexní SEO nástroje

Na internetu je dnes k dispozici výborná služba, která kromě validity stránek zkoumá také její SEO úroveň a boduje weby od 0-100 podle jejich kvality. Určitě vám tuto službu doporučuji. Je dobré prozkoumat komplexnost vašeho webu a porovnat ji s ostatními.

<http://seo-servis.cz/>

Hlavně je tato služba zdarma. Mnoho webových serverů se snaží nalákat zákazníky na optimalizaci SEO jejich webu za peníze. Většinou vám ale naučí práci pouze za to, co nyní v pohodě zvládnete sami. Takže si nenechte sosat šrajtofli ☺.

Shrnutí

- SEO je technologie optimalizace webových stránek pro vyhledávače a přístupnosti pro všechny uživatele.
- Rozlišujeme etické a neetické metody SEO.
- Součástí SEO je též validita zdrojového kódu.

Pojmy k zapamatování



SEO, Optimalizace, standardy, etické a neetické metody, Validita, Validátory

Samostatně

Pakliže by vás problematika SEO zajímala více do hloubky, můžete navštívit ještě následující odkazy.

Pro zájemce:

- <http://www.jakpsatweb.cz/seo/seo.html>
- <http://interval.cz/webdesign/seo-a-sem/>

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický

Průvodce studiem



Osmá kapitola se vázala k problematice SEO Optimalizace. V této fázi už bychom měli být připraveni pustit se do zveřejnění našeho díla na veřejném webovém serveru. Čeká nás tedy poslední kapitola, kde se podíváme na to, jak a kde můžeme naše webové stránky zveřejnit.



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

9 Publikování www

Cíle

Po prostudování této kapitoly budete:

- umět publikovat www stránky na internetu,
- využít služeb webhostingu.

Doba potřebná ke studiu

Na prostudování kapitoly Vám bude stačit **120 minut**.

Webhosting

Máte-li po nějakém čase stráveném nad svými webovými stránkami pocit, že jste dospěli do fáze, kdy byste je konečně rádi zveřejnili světu a publikovali na internetu, stojíte před základní otázkou.

Jak a kde je dostat na Internet a kde získat doménové jméno?

Otázce jak je dostat na internet se budeme věnovat později. Nyní si řekněme něco o tzv. **webhostingu**.

Webhosting je služba, kterou poskytují majitelé webových serverů, kteří nabídnou uživatelům prostor na svém serveru pro jejich webové stránky. Naše webové stránky tak vlastně hostují na cizím serveru – odtud slovo webhosting.

Existují dnes v podstatě dva základní typy webhostingů.

1. **Placený** – od 20 Kč / měsíc až po desítky tisíc měsíčně, podle nároků.
2. **Free** – hosting zdarma, ovšem za cenu reklamy a nižších kvalitativních a kvantitativních možností.

Sám webhosting by nám ovšem byl celkem k ničemu bez registrované domény:

např. www.chovatele-psu.cz, kterou by DNS server překládal na IP adresu serveru, kde naše stránky hostujeme. **Více jsme se tomuto tématu věnovali v první kapitole.**

O cenách domén si můžete zjistit více např. na <http://www.domena.cz/>



evropský
sociální
fond v ČR



EVROPSKÁ UNIE



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Firmy poskytující webhosting dnes pracují velmi komplexně a zajišťují tyto služby pohromadě. Chcete-li si tedy objednat webhosting zároveň s příslušnou doménou na jednom místě, není to problém. Vše vyřeší poskytovatel. Na vás je pouze rozhodnutí, jaký druh webhostingu zvolíte a zda vámi vybraná doména je volná. A samozřejmě kolik do celé záležitosti hodláte investovat.

V žádném případě nelze registrovat 2 shodné domény. Pochopitelné ale lze měnit doménové přípony. Lze např. registrovat *www.chovatele-psu.cz*, ikdyž už existuje doména *www.chovatele-psu.com* (uvádím jen příklady, určitě dnes existují obě dvě).

Je potřeba se dobře rozhodnout, jaké doménové jméno zvolíte, protože jeho pozdější změna s sebou přináší nemalé komplikace a často i další peníze.

Známí poskytovatelé placeného webhostingu u nás jsou např.

- <http://www.forpsi.cz>
- <http://www.superhosting.cz/>
- <http://www.czechia.com/>

Free webhosting

Pokud nemáte zájem investovat peníze do webhostingu, můžete využít služeb některého poskytovatele, který vám zajistí jak webhosting, tak doménu **úplně zdarma**. Má to pochopitelně svoje **ALE**.

1. Váš web bude doplněn automatickým vložením reklamy, která bude na vašich stránkách neustále umístěna (ve většině případů, neplatí ale vždy – např. php5.cz).
2. Získaná doména bude třetího řádu, tzn., že bude vypadat asi nějak takto: <http://vašejmeno.poskytovatel.cz> (v drtivé většině případů)

Pro příklad si předvedeme doménu <http://chovatele-psu.php5.cz>, kde **chovatele-psu** je náš název, **php5** je označení poskytovatele a **cz** je naše národní doména.

Zřízení free webhostingu



U poskytovatele **PHP5 Freehosting** si nyní ukážeme jaký je postup zřízení.

1. Přejděte na odkaz www.php5.cz
2. Pod záložkou **Podmínky** se seznamte s příslušnými pravidly tohoto poskytovatele.
3. Pod záložkou **Registrace** zadejte zvolené jméno webu. Jak jsem dříve řekl, dobře si jej rozmyslete. Místo mezer použijte pomlčku. Klepněte na tlačítko **Registrovat**.
4. V případě, že je doména již registrována, budete na to ihned upozornění a musíte provést změnu.
5. Pokud je doména volná, budou vám na zadaný email zaslány nezbytné údaje.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

- Po potvrzení pokračujte dle pokynů – na email vám bude zaslán potvrzovací kód, pomocí něhož se registrace ověří a dokončí (Vyžodována nově registrační sms ☺).
- Nakonec vám poskytovatel zašle přihlašovací údaje se vstupem do administrace, z nichž nás nejvíce budou zajímat připojovací údaje FTP, pomocí nichž přesunete své stránky (popř. další soubory) na server.

FTP server: Váš server

FTP user: Vaše jméno

FTP heslo: Vaše heslo

Tyto připojovací údaje si spolu s dalšími, které jste obdrželi po registraci emailem, dobře uložte, popř. vytiskněte.

FTP

Pro přenos souborů na internetu se využívá protokolu (služby) **FTP**. Stejně jako pro práci s protokolem HTTP se používají internetové prohlížeče, tak pro práci s FTP jsou k dispozici velmi jednoduché programy (tzv. FTP klienti).



Můžeme využít např. služby Total Commanderu anebo vám můžu vřele doporučit volně dostupný program **FileZilla**, který patří mezi ty nejlepší.

FileZilla – ke stažení (Stahujte verzi KLIENT)

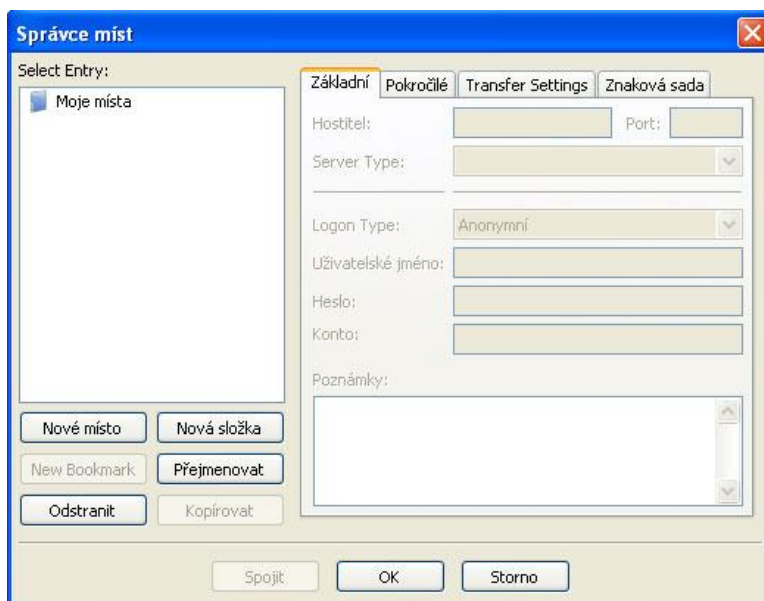
<http://filezilla.cz/>

Po stažení si program nainstalujte. Velmi jednoduše, pouze se řídíte pokyny instalace. Po instalaci se vám program spustí.

Klepněte na záložku **Soubor** a vyberte volbu **Správce míst**.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

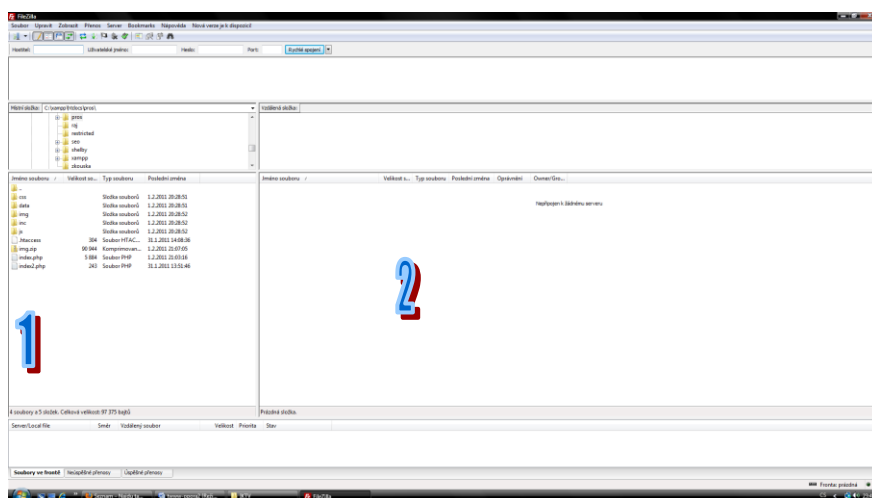
Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -
Kristýna Kubrická a Jan Kubrický



Obr. 20 Správce míst FTP klienta FileZilla

1. Klepněte na tlačítko **Nové místo**.
2. V seznamu míst se objeví Nové místo – pojmenujte ho dle svého uvážení, nejčastěji podle názvu domény.
3. Do kolonky hostitel napište přesně údaj (název FTP serveru), který jste obdrželi v registračním emailu.
4. V nabídce Logon Type vyberte volbu **Normální**.
5. Vyplňte Uživatelské jméno pro dané připojení (FTP user).
6. Vyplňte heslo (FTP heslo).

Dokončete stiskem tlačítka **OK**.



Obr. 21 Hlavní okno programu FileZilla

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Po připojení, které můžete vysledovat v horním rámu programu, můžete ze sekce označené modrou **1** přenášet data ze svého počítače na server. Výpis vašich dat uložených na serveru, uvidíte po připojení v okně **2**. Jestliže jste ještě žádná data nenahráli, okno je prázdné. Data ze svého počítače stačí přetáhnout pomocí levého tlačítka myši. Přenos souborů na server je graficky a početně znázorněn ve spodním rámu programu.

Jakmile dokončíte přenos souborů na server, a pakliže mezi soubory byl přítomen dokument pojmenovaný **index**, spusťte int. prohlížeč a po zadání URL adresy vašeho webu uvidíte publikovány své www stránky.

Pojmy k zapamatování



Webhosting, Free Webhosting, Doména, FTP

Průvodce studiem



A jsme na konci. Devátou kapitolou jsme završili sondu do tvorby www stránek a jednoduchých webových aplikací. Věřím, že jste na samém studiu našli hodně pozitivního ☺ a rozhodnete se i dále rozšiřovat své znalosti, kterých je pro tvorbu kvalitních www stránek potřeba opravdu hodně.

Hodně zajímavých věcí se nám sem bohužel nevešlo. Zde mě napadá především Redakční systémy, jako je např. WordPress. Ovšem na tuto část se podíváme určitě v rámci prezenčních cvičení nebo seminářů.

Použitá literatura

1. KUBRICKÝ, J. *Tvorba WWW stránek*. 2012. Studijní opora vzniklá v rámci projektu programu Koordinátor ICT / projekt DVPŠ OK.
2. WEMPEN, F. *HTML a CSS : krok za krokem*. Brno: Computer Press, 2007. ISBN 978-80-251-1505-3.
3. HAUSER, M. *HTML a CSS : velká kniha řešení*. Brno: Computer Press, 2006. ISBN 80-251-1117-2.
4. SCHAFER, S. *HTML, XHTML a CSS*. Praha: Grada, 2009. ISBN 978-80-247-2850-6.
5. LACKO, L. *PHP 5 a MySQL 5*. Brno: Computer Press, 2007. ISBN 978-80-251-1695-1.
6. NARAMORE, E. *PHP 5, MySQL, Apache*. Brno: Computer Press, 2006. ISBN 80-251-1073-7.
7. GRAPPONE, J., COUZIN, G. *SEO*. Brno, Zoner Press, 2007. ISBN 978-80-86815-85-5.
8. DOMES, M. *333 tipů a triků pro CSS*. Brno, Computer Press, 2011. ISBN 978-80-251-3366-8.
9. LACKO, L. *AJAX*. Brno: Computer Press, 2008. ISBN 978-80-251-2108-5
10. HOGAN P. Bryan. *HTML5 a CSS3*. Brno: Computer Press, 2011. ISBN 978-80-251-3576-1.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

11. MAKZAN. Programujeme hry v HTML5. Brno: Compuiter Press, 2012. ISBN 978-80-251-3731-4.
12. MARGORÍN, M. *jQuery bez předchozích znalostí*. Brno, CPress. 2011. ISBN 978-80-251-3379-8.

Poznámka:

Neocitované obrázky a tabulky jsou dílem autora. Použité loga jsou spojena vždy s vysvětlovaným tématem dané technologie a jsou součástí oficiálních stránek, na které je odkazováno. PrintScreeny vznikly použitím probíraného SW.

Příloha 1

APACHE, PHP, MySQL



Dříve, než se lze pustit do testování skriptů v jazyce PHP na lokálním počítači (zobrazovat stránky v prohlížeči), je potřeba disponovat lokálním webovým serverem, interpretem jazyka PHP a nejlépe databází MySQL.

Pro PHP je jako stvořený webový server **APACHE**, který spadá stejně jako PHP a MySQL mezi OPEN SOURCE (tedy s otevřeným zdrojovým kódem) a lze jej taky zdarma získat a využívat.

Všechny 3 zmíněné komponenty lze získat zvlášť a instalovat jak v distribuci pro Windows, LINUX či Mac OS. Jejich vzájemná konfigurace ale vyžaduje pokročilejších znalostí a proto není pro začátečníky zrovna nejvhodnější.

K dispozici jsou proto **kompletní instalační balíčky**, které provedou 3 kroky v jednom. Jeden z takových si blíže popíšeme.

XAMPP



XAMPP je dostupný ke stažení na adrese <http://www.apachefriends.org/en/xampp.html>. Je instalátorem s otevřeným zdrojem, který instaluje Apache, MySQL, PHP, Perl, phpMyAdmin a server FTP. Je vhodný jak pro LINUX, tak pro OS od Microsoft.

V době, kdy jsem naposled aktualizoval tyto řádky, byla k dispozici ke stažení pro Windows stabilní verze s označením **XAMPP Windows 1.8.3**. Pokud narazíte na novější, nic se neděje. Vše by mělo v pořádku fungovat. Stáhněte si tento balíček (nejlépe jako Installer). Instalátor je velmi intuitivní

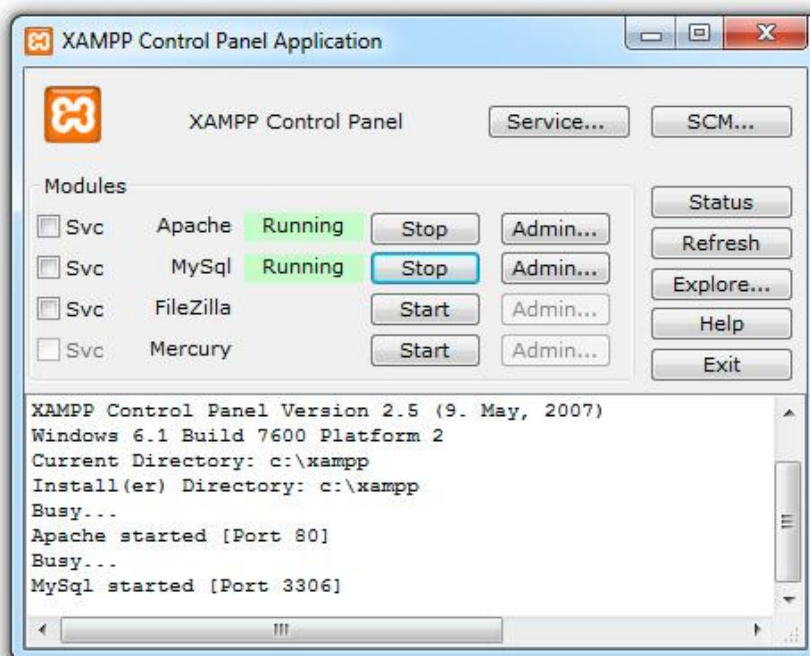
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

a neměli byste narazit na problém. Pokud možno zachovejte všechny defaultní volby, které vám instalátor nabídne. V případě, že byste měli problém s instalací (např. používáte ještě starší Windows XP, ve složce **Instalatory**, která je součástí cvičných souborů máte k dispozici jednu ze starších verzí balíku XAMPP, která je podporována i Windows XP.

Po dokončení instalace se vám spustí **kontrolní panel**, který signalizuje aktivní běžící služby. Tento kontrolní panel si můžete také spustit přímo z instalované cesty XAMPPu. Ve Windows je to stabilně C:\xampp\xampp-control.

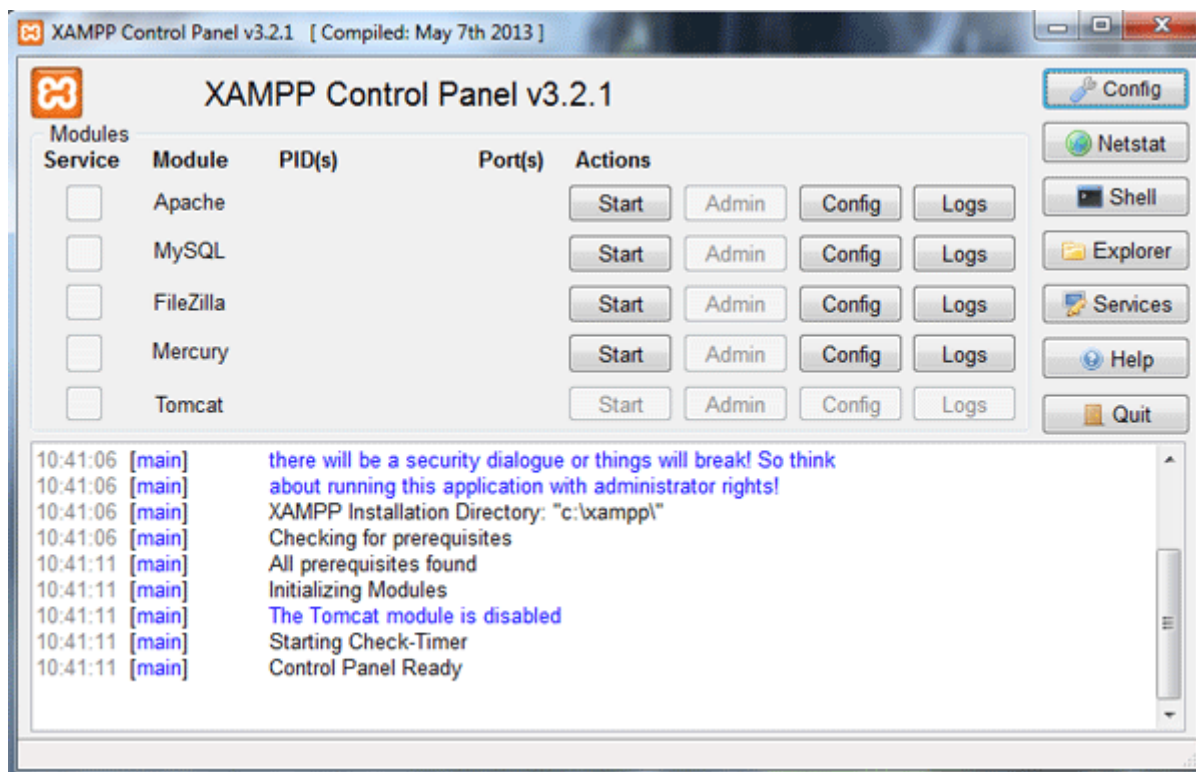


Obr. Kontrolní panel XAMPP verze 1.7.4

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický



Obř. Kontrolní panel XAMPP verze 1.8.2 (neplést s verzí Control Panelu)

Pro začátek nám bude stačit stisknout v Control Panelu tlačítko **Start** v řádku webového serveru Apache. Pokud se po uplynutí max. 10 sekund neobjeví zelené pole s textem **Running**, webový server se nepodařilo nainstalovat a budete muset hledat možné příčiny. Více v podkapitole této přílohy Problémy.

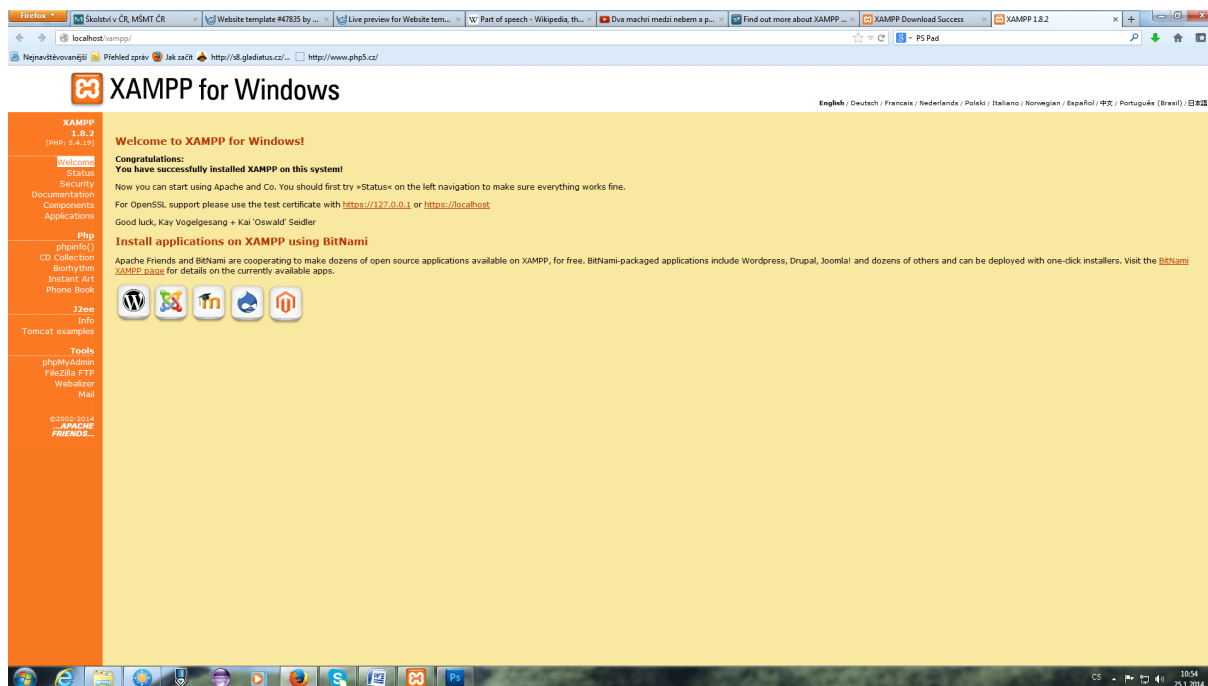
Pokud vám úspěšně běží webový server Apache, můžeme jej vyzkoušet přímo v internetovém prohlížeči. Otevřete si váš oblíbený prohlížeč a zadejte adresu URL ve tvaru: <http://localhost>

Pokud vše probíhá tak jak má uvidíte následující okno:

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický



Obr. Uvítací obrazovka XAMPP

Kořenový adresář pro dokumenty

PHP soubory nelze testovat v prohlížeči přímo z libovolného umístění např. na ploše vašeho OS, jak tomu bylo doposud, kdy jsme tvořili pouze statické stránky se značkovacím jazykem HTML, případně doplněnými JavaScriptem, který stejně jako HTML funguje v prohlížeči. Dokumenty obsahující PHP musíme spouštět skrze smyčku našeho lokálního webového serveru **localhost** (jeho IP adresa je 127.0.0.1).

Chceme-li tedy spouštět PHP skripty (nebo přímo www stránky obsahující i programový kód PHP), **musíme je nejprve umístit do kořenového adresáře serveru Apache pro webové dokumenty**. Tímto adresářem je v případě XAMPP složka **/htdocs/**. Ve Windows nejčastěji C:\xampp\htdocs.

V složce **/htdocs/** můžete libovolně vytvářet další podsložky. Na PHP soubor nebo stránku se pak v int. prohlížeči odkazujete podle cesty. Např. následovně:

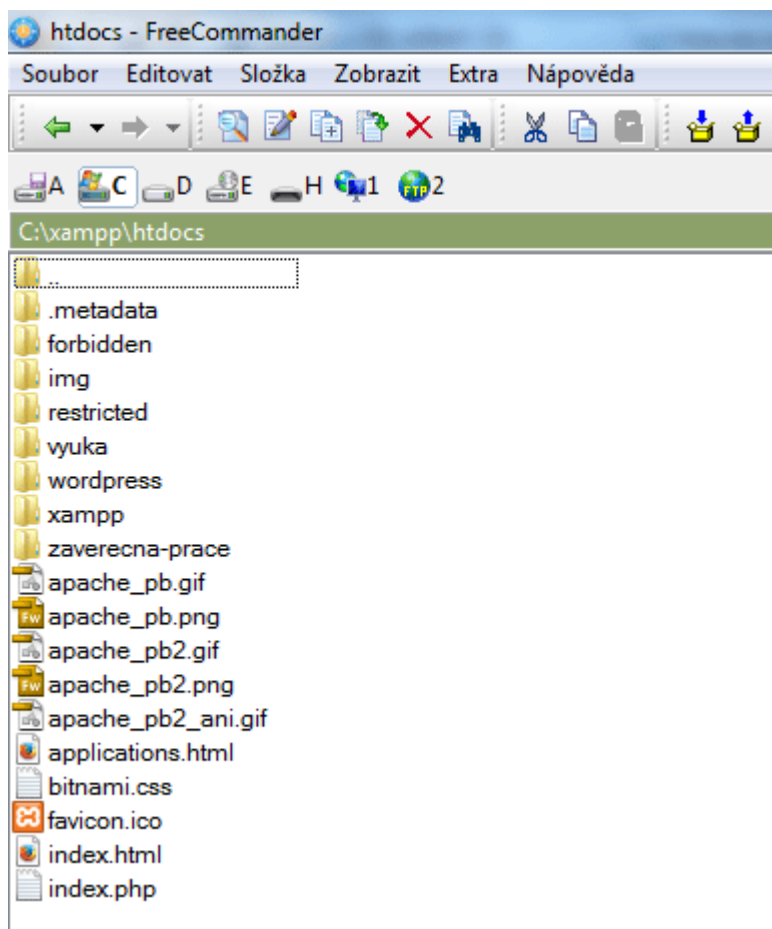
<http://localhost/vyuka/php/cviceni-3.php>

Tedy ve složce **htdocs** si vytvoříte složku **/vyuka/**, v ní složku **/php/** a v ní už budou ležet soubory, které si stáhnete pro potřeby studia ze 7 kapitoly. Není to nic složitého a věřím, že se s tím úspěšně poperete. Stejně bude ve složce **/htdocs/** ležet složka **/zaverecna-prace/**, která obsahuje rozpracovanou seminární práci, kterou máte k zápočtu dokončit.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický



Obr. Složka /htdocs/

Problémy

Při i po instalaci balíku XAMPP se mohou vyskytnout problémy. V případě nových verzí XAMPPu (od 1.8.n) a instalaci na Windows Vista či Windows 7 bude možná potřeba vypnout nástroj **Řízení uživatelských účtů** (UAC). Viz Ovládací Panely – Uživatelské účty. Instalace vás na tento problém upozorní.

Mohou vzniknout i jiné problémy související s kompatibilitou vašeho OS. V tomto případě raději instalujte starší verzi XAMPPu (v 1.7.4 ve složce Instalatory). Máte-li předtím nainstalovánu novější verzi, nejprve ji pomocí Uninstalleru odeberte.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Studijní opora: Základy tvorby www stránek a jednoduchých www aplikací -

Kristýna Kubrická a Jan Kubrický

Port 80

Webový server APACHE naslouchá defaultně na portu 80. V případě, že se vám Apache nedaří spustit, patrně na tomto portu běží již jiný program.

Může se např. stát, že na portu 80 koliduje spuštěný program **SKYPE** nebo server některého z programů Autodesk (např. Inventor), Firewall nebo jiný web server. Dříve než tedy Apache nastartujete, vypněte službu nebo program, který port 80 blokuje.

Jak zjistit, který program blokuje port 80?

Nápověda pro Windows

Otevřete příkazový řádek a zadejte příkaz **netstat -ano**.



```

C:\Users\Honza>netstat -ano

Aktivní připojení

Proto  Místní adresa      Cizí adresa      Stav      PID
TCP    0.0.0.0:80          0.0.0.0:0         NASLOUCHÁNÍ  202244
TCP    0.0.0.0:135        0.0.0.0:0         NASLOUCHÁNÍ  820
TCP    0.0.0.0:443        0.0.0.0:0         NASLOUCHÁNÍ  202244
TCP    0.0.0.0:445        0.0.0.0:0         NASLOUCHÁNÍ  4
TCP    0.0.0.0:554        0.0.0.0:0         NASLOUCHÁNÍ  3244
TCP    0.0.0.0:2869       0.0.0.0:0         NASLOUCHÁNÍ  4
TCP    0.0.0.0:5357       0.0.0.0:0         NASLOUCHÁNÍ  4
TCP    0.0.0.0:10243      0.0.0.0:0         NASLOUCHÁNÍ  4
    
```

Obr. Výpis programu netstat

Na obr. vidíme, že na portu 80 naslouchá nějaký program. Tento program má identifikátor procesu (PID) 202244. Velice jednoduše nyní ve **Správci úloh** najdeme program, který je pod tímto PID označen.

Nalezení ID procesu určitého programu

1. Stiskněte kombinaci kláves Ctrl+Alt+Del a poté klepněte na tlačítko **Správce úloh**.
2. Klepněte na kartu **Procesy**.
3. Pokud se nezobrazí sloupec PID, klepněte na nabídku **Zobrazit**, klepněte na příkaz **Vybrat sloupce** a poté zaškrtněte políčko **PID** (Identifikátor procesu).
4. Klepnutím na sloupec "PID" seřadíte uvedené procesy podle jejich identifikátorů. Podle zobrazených identifikátorů snadno naleznete názvy odpovídajících programů.